

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK
TABANLI BİR MODEL ÖNERİSİ**

YÜKSEK LİSANS TEZİ

Burak YARDIMCI

Bilişim Anabilim Dalı

Mimari Tasarımda Bilişim Programı

HAZİRAN 2012

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK
TABANLI BİR MODEL ÖNERİSİ**

YÜKSEK LİSANS TEZİ

**Burak YARDIMCI
(523091018)**

Bilişim Anabilim Dalı

Mimari Tasarımda Bilişim Programı

Tez Danışmanı: Öğr. Gör. Dr. Hakan TONG

30 MAYIS 2012

İTÜ, Fen Bilimleri Enstitüsü'nün 523091018 numaralı Yüksek Lisans Öğrencisi **Burak YARDIMCI**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK TABANLI BİR MODEL ÖNERİSİ**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Öğr. Gör. Dr. Hakan TONG**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. Dr. Gülen ÇAĞDAŞ**
İstanbul Teknik Üniversitesi

Doç. Dr. Hakan YAMAN
İstanbul Teknik Üniversitesi

Teslim Tarihi : **30 Mayıs 2012**
Savunma Tarihi : **06 Haziran 2012**

Sevgili anneannem Suriye Yardımcı'ya ithaf edilmiştir,

ÖNSÖZ

Tez çalışmam boyunca beni yönlendiren ve benden hiçbir zaman ümidini kesmeyen sevgili hocam Öğr. Gör. Dr. Hakan Tong’a, sevgi ve destekleri ile sürekli arkamda olan aileme, arkadaşlarıma ve tez çalışmamda bana her şekilde destek verip bu tezin gerçekleşmesini mümkün kılan sevgili dostum Dağhan Çam’a teşekkürlerimi borç bilirim.

Mayıs 2012

Burak Yardımcı
(Mimar)

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
ŞEKİL LİSTESİ.....	xiii
ÖZET.....	xv
SUMMARY	xvii
1. GİRİŞ	1
1.1 Tezin Amacı	2
1.2 Tezin Kapsamı.....	2
1.3 Tezin Yöntemi.....	3
2. TOPLU ÜRETİLEN KONUT TASARIM YÖNTEMLERİ.....	5
2.1 Toplu Üretilen Modüler Konut Kavramına Geleneksel Yaklaşımlar	5
2.2 Toplu Üretilen Modüler Konut Kavramına Ütopik Yaklaşımlar	11
2.3 Toplu Üretilen Modüler Konut Kavramına Bilgisayar Destekli Yaklaşımlar ..	13
2.4 Değerlendirmeler.....	25
3. MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK TASARIM TABANLI MODEL.....	29
3.1 Geliştirilen Modelin Amacı ve Tanıtımı	29
3.2 Modelin Algoritması	30
3.3 Modelin Arayüzü.....	31
Kontrol çubukları	32
Kütüphane sistemi.....	33
3.4 Modelin Çalışma Aşamaları.....	34
Çekirdek ve servis alanı	35
Temel modül	39
Daire tipolojileri	44
Dış taşıyıcı aksları	50
Model ve rhino programına transfer	51
Geliştirilen model.....	52
4. SONUÇ VE ÖNERİLER.....	61
KAYNAKLAR	65
EKLER.....	67
ÖZGEÇMİŞ.....	94

KISALTMALAR

BIM	: Building Information Modelling
SAR	: Foundation of Architects Research
SAAL	: Servico de Apoio Ambulatorio Local
MHAS	: Modular Housing Arrangement System
MIT	: Massachusetts Institute of Technology

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : The Scheepstimmemanstraat projesi (Url-1).....	6
Şekil 2.2 : The Scheepstimmemanstraat daire tipleri (Url-1).....	6
Şekil 2.3 : The Scheepstimmemanstraat (Url-1).....	7
Şekil 2.4 : Malqueira konutları – Alvora Siza (Url-2).....	7
Şekil 2.5 : Malqueira konutları – Alvora Siza (Url-2).....	8
Şekil 2.6 : Hudco Konutları – Modüler Türetme (Torus, 2008).....	9
Şekil 2.7 : Hudco Konutları – Modüler Türetme (Torus, 2008).....	10
Şekil 2.8 : Daire tipleri (Url-3)	10
Şekil 2.9 : Habitat 67 (Url-4).....	11
Şekil 2.10 : Cluster – City in the air (Kurokawa, 1993).....	12
Şekil 2.11 : Felix City (Kurokawa, 1993)	13
Şekil 2.12 : Vertical Village (Url-5)	14
Şekil 2.13 : Vertical Village (Url-5)	15
Şekil 2.14 : Çekirdek ve modül ilişkisi (Url-6)	16
Şekil 2.15 : Modül - Kütle (Url-6).....	16
Şekil 2.16 : Barcode Housing türetim (Url-7)	17
Şekil 2.17 : Kontrol çubukları ile üretilen türetmeler (Url-7)	18
Şekil 2.18 : Ara bölmelerle oluşan farklı iç mekanlar (Url-7).....	19
Şekil 2.19 : Yatay ve dikey sirkülasyonlar (Url-7).....	19
Şekil 2.20 : Kütle oluşumu (Url-7)	20
Şekil 2.21 : Daireler diyagramı – Cephe kurgusu (Url-7)	21
Şekil 2.22 : Üretim sistemi (Url-7)	22
Şekil 2.23 : MHAS süreç sistemi (El-Zanfaly, 2009).....	23
Şekil 2.24 : MHAS arayüzü (El-Zanfaly, 2009).....	24
Şekil 2.25 : İlk alternatif (El-Zanfaly, 2009)	24
Şekil 2.26 : İkinci alternatif (El-Zanfaly, 2009)	25
Şekil 3.1 : Modelin çalışma prensibi	30
Şekil 3.2 : Processing arayüzü-sekmeler	31
Şekil 3.3 : Kontrol çubukları	32
Şekil 3.4 : Farklı bakış açıları	34
Şekil 3.5 : Processing “Setup” ve “Draw” bölümleri	37
Şekil 3.6 : Çekirdek alanının oluşumu	37
Şekil 3.7 : Çekirdek alanı planı ve modeli	37
Şekil 3.8 : Karşılaştırmalı çekirdek alanları	38
Şekil 3.9 : Yüzeysel ve tel örgü çekirdek	39
Şekil 3.10 : Modeldeki daire alanları	40
Şekil 3.11 : İlk modül yerleşim alanları	41
Şekil 3.12 : Modül yerleşimi ve kesintisizlik	41
Şekil 3.13 : Modül büyüklük parametreleri	42
Şekil 3.14 : Modül büyüklüğü	43
Şekil 3.15 : Modül birleşim şekilleri	43

Şekil 3.16 : Fizibilite-Tasarım şeması	44
Şekil 3.17 : Tiplerdeki modül değişimleri	45
Şekil 3.18 : Tiplerin dağılım oranları.....	46
Şekil 3.19 : Tiplerdeki renk değişimleri	46
Şekil 3.20 : Tiplerdeki modül sayısı değişimleri	47
Şekil 3.21 : Daire yoğunlukları.....	48
Şekil 3.22 : Daire alternatifleri	49
Şekil 3.23 : Daire plan çözümleri	49
Şekil 3.24 : Genel kat planı.....	50
Şekil 3.25 : Taşıyıcı sistemleri.....	51
Şekil 3.26 : Dışa aktarım.....	52
Şekil 3.27 : Rhino-Çekirdek alanı betiği	52
Şekil 3.28 : Rhino-Çekirdek alanı oluşumu.....	53
Şekil 3.29 : Rhino-1.tip daireler.....	53
Şekil 3.30 : Rhino-2.tip daireler.....	54
Şekil 3.31 : Rhino tüm daireler.....	54
Şekil 3.32 : Model-3ds max.....	55
Şekil 3.33 : Çekirdek - Modüller	55
Şekil 3.34 : Cephe elemanları	56
Şekil 3.35 : Tüm cephe kurgusu	56
Şekil 3.36 : Taşıyıcı ve teraslar.....	57
Şekil 3.37 : Cephe malzemesi.....	57
Şekil 3.38 : Modele ait kolajlar.....	58
Şekil 3.39 : Hücresel taşıyıcı sistem	59

MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK TABANLI BİR MODEL ÖNERİSİ

ÖZET

Özellikle 2. Dünya Savaşı sonrası doğan konut ihtiyacı ile ortaya çıkan toplu konut kavramı beraberinde standartlaşmayı getirmiştir. Kullanıcısının gereksinimlerini dikkate almadan, herkes için aynı tip ve yaklaşımla yaratılan yaşam alanları, daha sonraları hem mimari hem de kentsel anlamda sorgulanmaya başlanmış ve tasarımcıları alternatif tasarım yaklaşımlarına yönlendirmiştir. Tez çalışmasında 1960'lı yıllardan günümüze kadar olan süreçteki toplu konut tasarımında kullanılan alternatif yöntemler örnekler üzerinden incelenmiş ve alternatif bir tasarım modelinin nasıl bir yapıda olması üzerinde durularak yeni bir model yaratılmıştır.

İkinci bölümde, ilk olarak alternatif toplu konut tasarımı yöntemleri üzerinde durulmuş ve farklılıkları ile benzerliklerine değinilmiştir. Üzerinde durulan bu farklı yöntemlere örnekler üzerinden yaklaşılmıştır. Örnekler üzerinden değerlendirmeler yapılmış ve yaklaşımlar ortaya konulmuştur. Her örnek bazında, tasarlanacak model çerçevesinde dikkate alınacak noktalar belirlenmiş ve gerektiği yerlerde eleştiriler ve eksik taraflar da belirtilmiştir. Alternatif toplu konut tasarımları geleneksel, ütopyik ve bilgisayar destekli olmak üzere üç bölümde toplanmıştır. Her bölüm için farklı tarzda örnek projeler araştırılmıştır. Her projeden gerekli alanlar model için referans olarak alınmış ve model yapısında işlenerek değerlendirilmiştir.

Üçüncü bölümde daha önce kritiği edilen örnekler de ele alınarak tasarlanan model anlatılmıştır. Tasarım modelini oluşturan parametreler, koşullar ve modelin diğer tüm özellikleri üzerinde durulmuştur. Model aracılığıyla elde edilen toplu konut projesinin tüm aşamaları anlatılmıştır. Modelin yaratılması süresince ortaya çıkan problemler, bu problemlerin nasıl aşıldığı, arayüzün işleyiş şeması ve son ürüne dair datalar detayları ile anlatılmıştır. Modelin yaratılmasında kullanılan program ve bu programdan elde edilen ürünün görselleştirilmesi üzerinde durulmuştur. Kullanıcının belirlediği parametrelerle gelişen ve gerçek zamanlı değişimleri gözlenen model ayrıntılı olarak, betik yapısı ile birlikte işlenmiştir. Modelin belirlenen kullanıcı senaryoları üzerinden oluşan plan şemaları oluşturulmuştur.

Son bölümde ise tasarlanan modelin hangi yönlerden eksik kaldığı üzerinde durulmuştur. Başlangıçta belirlenen hangi hedeflere ulaşıldığı, hangilerine ulaşamadığı belirlenmiştir. Gelecekte modelin geliştirileceği alanlar belirlenmiş ve ne yönde evrimleşmesi gerektiği üzerinde durulmuştur. Bu tür modellerin alternatif konut tasarımıdaki rolünün önemine değinilerek tez tamamlanmıştır.

A PARAMETRIC MODEL PROPOSAL FOR MODULAR HOUSING DESIGN

SUMMARY

A wide-range housing problem was arisen especially after First World War all around the world. To solve that problem, new housing reproduction era started in all cities that had lost the housing substance after the war. The standardization method had used to solve the need of that housing substance. The standardization approach that had developed with the Ford in motor industry called as T-model production had applied to the architecture field to solve the housing demand in a short time. Although this approach of standardization was successful to solve the problem of huge housing demand, it has caused negative effects on habitants.

The new problematic community who has different needs and expectations even so living in same type of flats had appeared. The housing problem was caused by standardization forced the designers to search for a new method of housing design especially after 1960's. The new alternative methods that were used in housing design are read in that research by examples. The examples are criticized with their advantages and disadvantages while creating a new model of modular housing design.

In the second chapter of the thesis, the different methods that are used in housing design after 1960's are exemplified. The methods are researched by their examples that are divided in three parts as conventional methods, methods in utopian projects, and computer aided housing designs. All examples are criticized with their effects, advantages or disadvantages. The possible parts of the projects that will use in the model are explained with their reasons and the general explanation is made by all examples.

The Scheepstimmermanstraat projects that were first planned by Jon Habraken then, designed by West 8 group, Malaqueira houses by Alvora Siza, Charles Correa's Hudco project and Habitat 67 housing project by Moshe Safdie are researched as examples for conventional methods in housing design.

Common positive characteristic of conventional examples are modular way of approach. In all projects, modular derivation method is used to reach geometries that are more complex with simple structures. Another positive characteristic that existed only in Malaqueira project is the design process that related to user need and time flow. The common negative characteristics of conventional examples are the limited creativity in projects that have only variability in planning scheme and facade design.

Another method that is used for housing design can be seen in utopian projects during 70's. Especially, the projects from Kisho Kurokawa, Arata Isozaki and Kenzo Tange are detailed criticized in that chapter. The main common point in all utopian projects is to see the building as an organical life form that grows by prescribed scheme. The designers tried to create living habitats in all projects. They planned a core that contains all service systems, infrastructures and vertical circulation system

inside and placed the living units at the edges of that core. The complexity of all service systems and structure is solved by that core system easily. However, the negative characteristic of utopian projects is the limited creativity in living units. All the living units are just repetition of themselves and there is not any variability in living modules.

The last method is used for housing design, that is searched in thesis is computer aided design method. Computer aided design method was firstly started as non-parametrical projects during 90's. The Vertical Project that was designed by MVRDV group is read as first example in thesis. T-trees Social Housing project is second example to understand the modularity in evolution. Other examples, which are discussed for computer aided design method, are Barcode project and MHAS Modular Housing Arrangement System.

Nearly all examples are based on the modularity approach as the conventional housing projects. However, with the development of CAD systems the number of the parametric projects is increased.

The aim of reaching variability by the evolution of modules is common specialty in projects that are developed with computer aided design method. Designers tried to reach geometries that are more complex by starting from one module and they defined several living units. Generally, in all examples, the relation between living areas and core system has important role. Designers tried to place the core system first by defined rules and then try to reach geometries environs of a core. Another characteristic of computer-aided systems is their interface. Some models are based on real-time change interface although some of them are not. The real-time change interface specialties criticized positively and try to adapt in new model that will create. The algorithms of all examples are examined carefully and try to find the similar way of algorithm using in new model.

In the third chapter, the new model is defined with its details in all phases. The reason of choosing the program is explained and the program that used to create the model was being introduced with its characteristics, interface and work scheme. The characteristic of the program is classified and the reasons during the selection of the program are told. Other programs and work schemas was presented with the reasons for choosing them.

All the parameters, circumstances and all other specialties of the model are explained with their causes and results. The workflow of the model is shown with images and all problems that are existed during the process are defined. All the parts of the model such as sliders, parameters, export and library specifications and interface are explained and the real time changes are shown with images. The role of the parameters is explained with their ranges and numeric specialties. The algorithms of the model are written in that part with explaining the codes and parameters for relevant researchers. The optimal values and their reasons are explained in that part for parameters.

The parts of the script are explained with visualizations that are prepared by other programs. The export and import relations of the program with other CAD programs are introduced. The project that was created by the model was converted to the other data files to complete presentation. After that, the project was visualized and the project was completed.

In the last part of thesis, the new model is criticized with its negative and positive characteristics. The deficient points of the model were explained with their reasons and the goals for future researches are defined. The possible evaluation of the model in near future is explained clearly and the boundary of the parametric model is drawn. The possible areas of expertise and usage of the model is being told. The studies that can be develop after the thesis project and its prospective future is discussed. Future targets of the model was explained with their target aims was explained.

The missing parts of the model such as database creation, data mining process, variability in core parameters and the visualization process is written with their roles in near future works. The way of attachment of these properties to the future model is explained clearly.

The thesis project concluded with the explanation of the importance role of the similar models in housing design process and their potential effects during the housing design process. The importance of the variability of the last product that is created by model algorithm is discussed with architectural prospective.

1. GİRİŞ

Konut, tasarım ihtiyacının insanın gündelik hayatına nüfuz etmeye başladığı ilk günden beri, her zaman sadece tasarımcının veya mimarın değil aynı zamanda onu kullanan kullanıcının da tasarlamaya ve inşa etmeye istekli olduğu bir yapı türüdür. Aslında diğer tüm mimari problemlerin ve konuların içinde tasarımcısı veya uzmanı dışında kullanıcısı tarafından tasarlanan tek yapı birimi olarak sınıflandırılabilir. İnsanın kendindeki bu, yaşam alanını tasarlama ve isteğince kullanma iç güdüsü, konut mimarlığının tarih sürecindeki pratik uygulamalarına yeterince yansıyamamıştır. Fakat teorik anlamda tasarım ve kuramcılar kullanıcısının belirlediği sınırlar çerçevesinde gelişen ve oluşan bir konut mimarisi mümkün mü sorusunu sürekli sormaya yöneltmiştir. Mimari tasarım sürecinde bilgisayar destekli tasarım anlayışının gelişmesi ve BIM destekli sistemlerin çoğalması ile, her bir kullanıcı için farklı bir tasarım ürünü yaratmanın önündeki karmaşık bilgi işleme süreci ortadan kalkmış ve konvensiyonel tasarım sürecine alternatif kullanıcı özelliklerine hitap eden bireysel tasarım sürecini kaçınılmaz hale getirmiştir.

20. yüzyıl'ın başlarında Henry Ford'un Ford T modeli için geliştirdiği seri üretim konsepti tüm dünyadaki üretim sistemini değiştirmiş ve sistematize edilmiş bir üretim anlayışını doğurmuştur. Daha sonraları 1920'lerde bu paradigma konut sorununu çözmek için mimarlığa aktarılmıştır (Benros & Duarte, 2008). Walter Gropius tarafından Almanya'da gerçekleştirilen Törten ve Le Corbusier' in Pessac konutları endüstrileşmeyi kullanarak maliyetleri azaltıp, ekonomik konut çözümleri geliştirmeyi amaçlamıştır (Kupferberg, 1974). Daha sonraları özellikle II. Dünya Savaşı sonrası kentlerde doğan konut ihtiyacını karşılayabilmek için seri üretim yöntemi tüm hatlarıyla benimsenmiştir. Konut pazarı üzerine yapılan bir takım araştırmalar sonucu kabul edilen genel doğrular tüm konut projelerinde benzer şekilde gerçekleştirilmeye başlanmıştır. Fabrikasyon üretim sistemi ve konvensiyonel tasarım süreçleri hem proje masraflarını azaltmış hem de savaş sonrası açığa çıkan konut ihtiyacının giderilmesini sağlamıştır. Kullanıcısı için özelleşmiş toplu üretilen konut tasarımı yaklaşımının gelişmesi de aslında tam da bu

seri üretim anlayışının en yaygın olduğu döneme rastlamaktadır (Benros & Duarte, 2008).

1960'lı yıllarda Jon Habraken'ın içinde bulunduğu SAR grubunun Hollanda'daki çalışmaları ile başlayan (Benros & Duarte, 2008) kullanıcısı çevresinde değişen konut tasarımı anlayışı günümüze kadar farklı yöntemlerle teori ve pratikte denenmiştir.

1.1 Tezin Amacı

Bu çalışmadaki amaç; 1960'lı yıllardan itibaren geleneksel konut tasarım mantığı dışında gerçekleştirilen tasarım yöntemlerini örnekler üzerinden irdelemek ve bu yöntemleri rasyonel ve irrasyonel yanlarıyla ortaya koymaktır. Bu yöntemleri örneklemelerle analiz edip, çıkarsamalar yapmak ve yapılan bu çıkarsamalarla yeni bir alternatif yol önermektir. Yapılan çalışmaların hem olumlu hem olumsuz taraflarını irdeleyip bunlar üzerinden beslenerek ortaya koyulan bir çalışma ile konut tasarımını kullanıcı özelinde şekillendiren bir model oluşturmaktır. Yaratılan bu modeli algoritmasıyla ortaya koyarak, bu algoritmayı oluşturan parametreleri seçim nedenleri ve etki alanlarıyla anlatmak ve çalışma prensibini açıklamaktır. Yazılımdan alınacak ürünleri görsellerle tanımlamak ve en sonunda tüm sistemin bir özeleştirisini yaparak eksik yanlarıyla ileride nasıl geliştirilebileceğini ortaya koymak ve olumlu yanlarıyla tasarım sürecine ne tür katkılar sağladığını anlatmaktır.

1.2 Tezin Kapsamı

Tez kapsamında 1960'lı yıllardan itibaren günümüze kadarki olan sürede geleneksel yöntemler, ütöpik projeler ve bilgisayar destekli çalışmalar ile gerçekleştirilmiş toplu üretilen konut projelerine örnekler üzerinden değinilmiştir. Bu projelerin işleyiş şemaları, getirdikleri yenilikler, tasarım süreçlerinde kullanılan yöntemler ve kullanıcı özellikleri çerçevesindeki ele alınış biçimleri tartışılmıştır. Çalışmada sadece toplu üretilen ve birden fazla kullanıcısı olan konut tasarımı başlığı üzerinde durulmuştur. Bu sayede aynı proje bünyesinde birçok ihtiyaç girdisine ne denli cevap verilebileceği olasılığı işlenmiştir. Örneklemelerden sonraki bölümde, örnekler üzerinden yapılan değerlendirmeler ve diğer mimari yaklaşımlar ile yeni bir model oluşturulmuş ve modelin çalışmasındaki parametreler üzerinde durulmuştur.

Yaratılan modelin görsel olarak zengin bir proje üretmesinden ise toplu üretilen konut projelerindeki daire şemalarını kullanıcısına bağlı olarak üretmesi hedeflenmiştir. Birçok alternatifte üretilen konut tipolojilerinden sonra, sonuç ürünlerin nasıl diğer yazılımlara aktarılabilceğı ve işlenebileceğı tartışılmıştır.

1.3 Tezin Yöntemi

Tez genelinde izlenen yöntem ilk olarak örneklemlerle tasarım kararlarının ele alınması ve daha sonra yazılacak algoritma ile de modelin oluşturulması olarak iki başlıkta irdelenebilir. Bu bağlamda; İlk olarak tez konusu belirlenmiş ve tezin hedefleri ortaya konmuştur. Tez konusu olarak kapsam bölümünde anlatıldığı gibi ilk olarak 60'lı yıllardan bu yana gerçekleştirilen bir takım projeler örneklenmiştir. Bu projelerin hangi tasarım yöntemleri ile oluşturuldukları, kullanıcıları ile nasıl bir etkileşim içerisinde bulundukları ve tasarım süreçlerindeki yaşanan sorunları üzerinde durulmuştur. Örneklemler üzerine yapılan analizlerden sonra ise yeni bir tasarım modelinin oluşturulması üzerinde çalışılmıştır. Modelin oluşturulmasına başlanırken hangi programın yaratılacak modelin tasarımı için daha uygun olduğu irdelenmiştir. Bu aşamada arayüzü basit olan ve kolay bir şekilde algoritmanın yaratılabilceğı bir program olan Processing programı kullanılmıştır. Programda yaratılan betik ile kullanıcıların ihtiyaçlarına göre değişen farklı parametreler belirlenmiş ve bu parametrelerin değişimlerinin gerçekleştirileceğı arayüz oluşturulmuştur. Bu aşamada modelin gerçek zamanlı değişimlere açık olması ve değiştirilen parametrelerle sürekli yeni tasarım ürünlerinin izlenebilmesi sağlanmıştır. Modelden elde edilen tasarım ürününün mümkün olduğunca mimari karakterde görselleştirilmesine önem verilmiştir. Bu amaç doğrultusunda da başka programlar ile sorunsuz şekilde çalışabilecek bir ürün hedeflenmiştir. Betik, modelin rhino programına transfer edilebilmesine olanak verecek şekilde oluşturulmuştur. Tasarım ürünü bu programa aktarılmış ve bu programdan da 3ds Max formatına çevrilmiştir. Bu kısımda modele cephe kurgusu atanmış ve mimari görselleştirmeler bu program üzerinden tamamlanmıştır. Tasarım ürününün elde edilmesi ve daha sonra görselleştirmelerin tamamlanması ile sonuç bölümü ele alınmıştır. Bizzat modelin kendisi tarafından oluşturulamayan eksik alanlar belirlenmiş ve genel olarak tasarlanan modelin ileride hangi doğrultuda gelişmesi gerektiğı üzerinde durulmuştur. Bu eksik noktalar hedefler olarak belirlenerek bu hedeflere nasıl

ulařılacađı zerinde durulmuřtur. Tezin geneli hakkında deđerlendirmelerde bulunulmuř ve eleřtriler getirilerek tez sonlandırılmıřtır.

2. TOPLU ÜRETİLEN KONUT TASARIM YÖNTEMLERİ

1960'lı yıllardan günümüze kadar olan ki süreçte kullanıcı özelinde farklılaşan konut tasarımı yöntemleri bir doğrusal zaman çizgisi üzerinde ele alınabilir. Bu yöntemler ilk olarak geleneksel tasarım anlayışının farklı yorumlandığı fakat gelenekselden kopmayan örneklerle başlamıştır. Daha ileri tarihlerde, özellikle 1970'li yıllarda ütöpik projeler ve mega strüktürler üzerine şekillenen projeler ortaya koyulmuştur. Mimari tasarım sürecinde bilgisayar destekli tasarım anlayışının gelişmesi ile de bilgisayar destekli tasarım yöntemleri ile oluşturulan ve parametrelere dayanan projeler oluşmaya başlamıştır. Başlarda shape grammar yöntemi ile başlayan çalışmalar daha sonraları birçok parametreye bağlı değişen ve gerçek zamanlı tepki veren betikler ile devam etmiştir.

2.1 Toplu Üretilen Modüler Konut Kavramına Geleneksel Yaklaşımlar

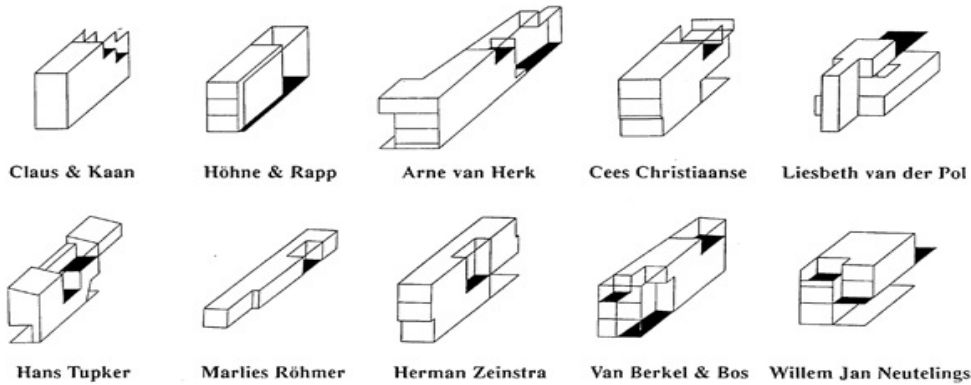
Toplu üretilen konut tasarımı üzerine geliştirilen yöntemler ilk olarak 60'lı yıllardan sonra ortaya çıkmış ve bilgisayar destekli tasarım teknolojilerinin gelişmeye başladığı 90'lı yıllara kadar pratikte ve teoride ve bazen de ütöpik çalışmalarla süregelmiştir.

Jon Habraken'ın Rotterdam şehri için SAR çalışma grubu ile ortaya koyduğu "INFILL / SUPPORT" projesi geleneksel yöntemler için ilk çalışmalardan biri olarak gösterilebilir. Proje "INFILL" ve "SUPPORT" terimleri üzerine oluşturulmuştur. "SUPPORT" proje genelindeki rijit, sabit kalan taşıyıcı ve altyapı sistemlerini tanımlamaktayken, "INFILL" ise projedeki esnek ve değişken kısımları anlatmaktadır. Bu konut üretim yaklaşımının en çarpıcı örneği Rotterdam'daki eski bir liman alanı üzerinde oluşturulan (Şekil 2.1/2.2/2.3) The Scheepstimmermanstraat toplu konut projesidir (Benros & Duarte, 2008).



Şekil 2.1 : The Scheepstimmestraat projesi (Url-1).

Planlaması West 8 grubu tarafından yapılan projedeki amaç, oluşturulan bir tartan grid üzerine inşa edilen, taşıyıcıları ve altyapıları değişmeyen fakat diğer tüm mimari elemanları kullanıcılarına göre değişim gösterebilen bir toplu konut alanının planlanmasıdır (<http://www.west8.nl>). Toplamda 2500 konuttan oluşan projede tümü 3 kat üzerine teras bahçeli olacak şekilde ve tümü birbirinden farklı cephe karakterine sahip projeler oluşturulmuştur. Şekil 2.2 de görülebileceği gibi Habraken'ın planlamasındaki taşıyıcı ve altyapı şeması değişmeden her kullanıcı için farklı tipte bir konut tasarımı oluşturulmuştur.



Şekil 2.2 : The Scheepstimmestraat daire tipleri (Url-1).

Projede taşıyıcı sistemin önceden bir sistematik içerisinde değerlendirilmesi ve tüm altyapı elemanlarının aynı bakış açısı ile yapım aşamasında çözülmesi olumlu iken, sadece cephe kurgusunda çeşitlenmenin yaratılabildiği olması tekdüzeliğe neden olmuştur. Bu projede ele alınarak program dahilinde kullanılması gereken nokta sistematik bir şekilde çözümlenmiş taşıyıcı altyapısıdır.



Şekil 2.3 : The Scheepstimmestraat (Url-1).

Kullanıcılar üzerinden gelişen konut tasarım anlayışının 70'lerde yaygınlaştığı diğer bir ülke ise Portekiz'dir. Bu yıllarda konut ihtiyacına çözüm bulmak üzere devlet tarafından kurulan SAAL grubunun başına Alvora Siza getirilmiş ve 1974 Karanfil Devriminin ardından değişen siyasi ortamın da katkısıyla, konutları kullanacak olan işçi sınıfının ihtiyaçları göz önüne alınarak yeni bir konut planlamasına yönelinmiştir. Evola kentindeki Malaqueira konutları projesinde 8 m. x 12 m. büyüklüğündeki arsalara planlanan farklı plan şemalarındaki konut birimlerinin, değişen topoğrafyaya da uyum sağlayarak farklılaşması sağlanmıştır. Temelde ön ve arka tarafta avlulara sahip olan iki farklı plan kurgusunun birbirleri ile olan birleşimleri çeşitli konut tipleri oluşturmuştur (<http://housingprototypes.org>)(Şekil 2.4/2.5).



Şekil 2.4 : Malqueira konutları – Alvora Siza (Url-2).

Sırt sırta veren bu iki tip daire şemasının ikisinde de yaşam ve yemek mekanları avlu kotunda olup açık merdivenler ile yatak odalarına ulaşmaktadır. Bu iki tip doluluk ve boşluk oranlarının zemin katlardaki giriş boşluklarının yüksekliklerinin ve diğer açıklık oranlarının her konutta farklılaştırılması ile sürekli çeşitlendirilmiştir. İlk başta tek katlı ve iki odalı konutlar olarak planlanan tiplerin daha sonraları kullanıcıları tarafından ihtiyaçları doğrultusunda büyütülmesine olanak sağlanmıştır. Bu sayede yıllar süresince bazı birimler tek katlı kalırken bazılarındaki kat sayısı artmış, yeni teraslar ve odalar eklenerek her birimin birbirinden farklılaşması sağlanmıştır. Tüm bu oda yerleşimleri kullanıcıların seçimine göre plana işlendiği için toplamda birbirinden farklı alternatifte yüzlerce konuta ulaşılmıştır (<http://alvorasizavieira.com>). Projenin tasarımında altyapısal bazda yerleştirilen kurallar çerçevesinde, yıllar geçtikçe gereksinimlerden kaynaklanarak yapılan eklemeler sayesinde projenin daha fazla oranda çeşitlenmesinin de önü açılmıştır (Benros & Duarte, 2008).

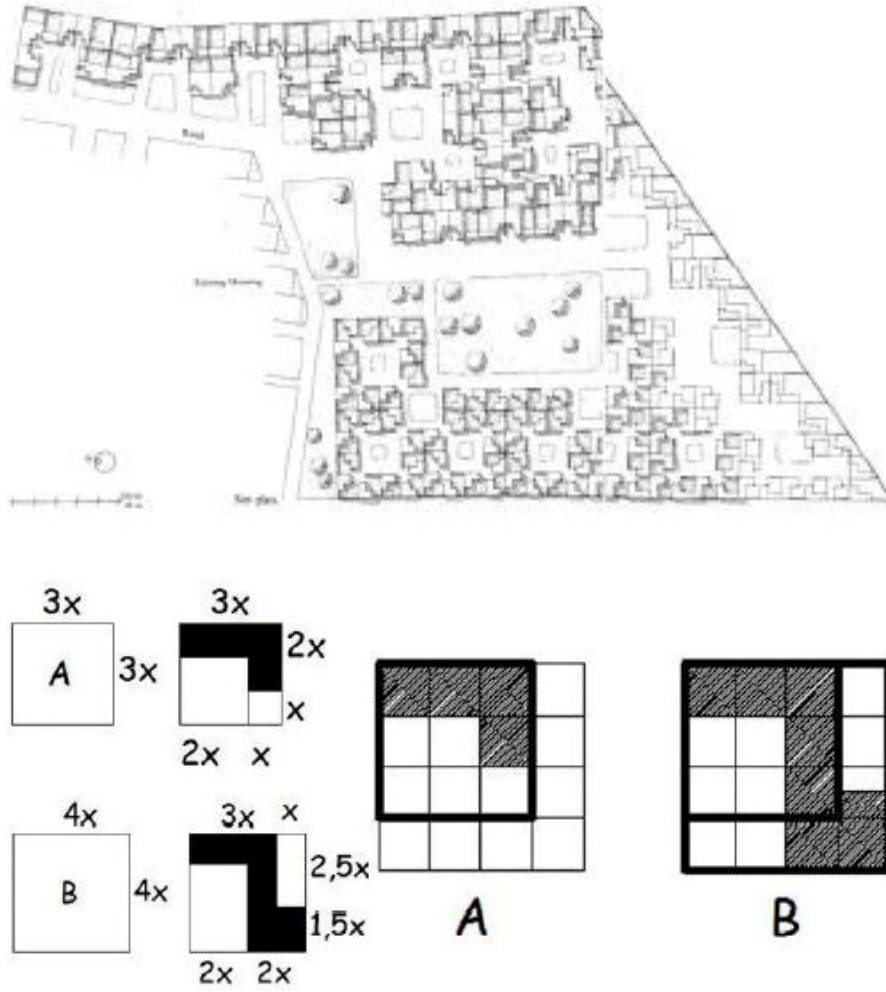
Proje genelinde, yaratılacak betik içerisinde kullanılabilecek en önemli nokta zamana dayalı bir çeşitlenme olasılığının kullanıcıya bırakılmış olması ve bu sayede çeşitlenmenin sınırsız bir zaman dilimine yayılmış olmasıdır. Yazılacak betik içerisinde de buna benzer bir zaman akışına gidilip konut sisteminin büyüyerek çoğalması veya küçülerek yok olmasına imkan verilebilir.



Şekil 2.5 : Malqueira konutları – Alvor a Siza (Url-2).

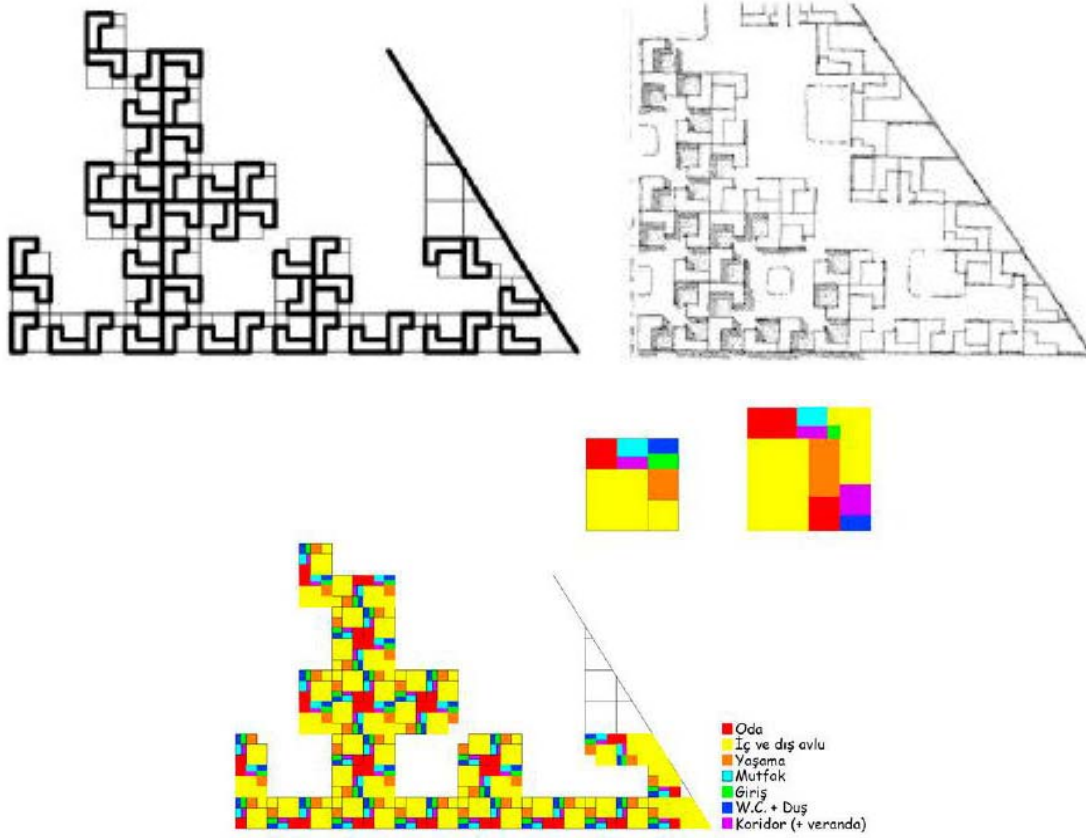
Geleneksel yöntemlere örnek teşkil edebilecek diğer bir proje ise Charles Correa'nın gerçekleştirdiği HUDCO konut projesidir (Şekil 2.6/2.7). Türetme mantığı ile gelişen bu projede temel iki farklı boyuttaki (3x3 – 4x4 metre) modüllerin bir araya

gelmesi ile birimleri oluřturması ve bu birimlerin de kendi aralarında bir araya gelmesi ile farklı türlerde oturma řemaları ortaya ıkarmasıdır (Torus, 2008).



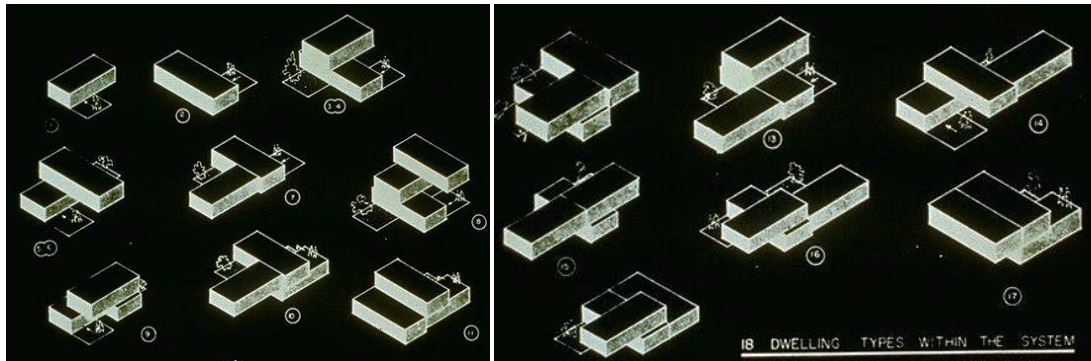
řekil 2.6 : Hudco konutları – Modler tretme (Torus, 2008).

Tretme kuralı drt farklı modln bir araya getirilerek ve 90 derece dndrlerek oluřturulmuřtur. Hudco projesinde her konut alanı iin bir yerleřim dzlemi belirlenmiř olması ve bu dzleme ebatları belirlenen modln tretilerek yerleřtirilmiř olması yazılacak betik iin rnek teřkil etmektedir. Bu sayede her yařam alanının minimum ve maksimum boyutları belirlenmiř ve modllerin belirli bir kurala gre remesi ile de eřitlilik saęlanmış olacaktır.



Şekil 2.7 : Hudco konutları – Modüler türetme (Torus, 2008).

Geleneksel yöntemlerle tasarlanmış toplu üretilen konut projelerine dair son örnek Moshe Safdie'nin Habitat 67 konut projesidir. Projede modüler olarak belirlenen birimin farklı şekillerde üretilmesi ve çoğaltılması ile büyük modüller elde edilmiştir. Toplamda 16 farklı prekast modül biraraya gelerek 158 konut birimini oluşturmaktadır (Şekil 2.8)(Url-3).

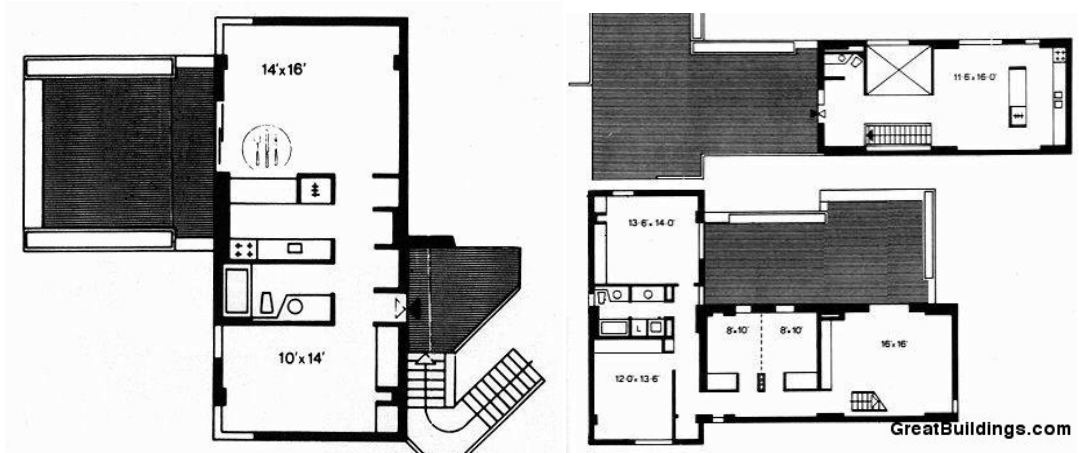


Şekil 2.8 : Daire tipleri (Url-3).

Projede modüllerin biraraya gelmeleri ile farklı alternatiflerde kapalı, açık alanlar ve teraslar yaratılmıştır. Sirkülasyon alanı yapının bir cephesinde kurgulanırken, ön cephesinde yaşam birimleri konumlandırılmıştır. Şekil 2.9' da görülebileceği gibi

farklı alternatiflerdeki modüllerin bir araya gelmesi ile aslında yaratılan doku bir konut bloğundan ziyade bir kent yapılaşmasını andırmaktadır (Url-3).

Habitat projesinde ele alınacak en önemli özellik sirkülasyon ve yaşam alanlarının birbirinden net bir şekilde ayrılmış olmasıdır. Bu sayede modüllerin çoğaltılmasında sirkülasyon alanı ile kesişimlerden ortaya çıkacak sorunlar aşılmıştır. Ayrıca proje başlangıcında aynı modülden üretilen 16 farklı birimin yaratılmış olması ve bu birimlerin çoğaltılması plan kurgusunun çözümünde esneklik sağlamıştır. Projedeki bu iki özellik yazılacak betiğe de referans teşkil etmektedir.



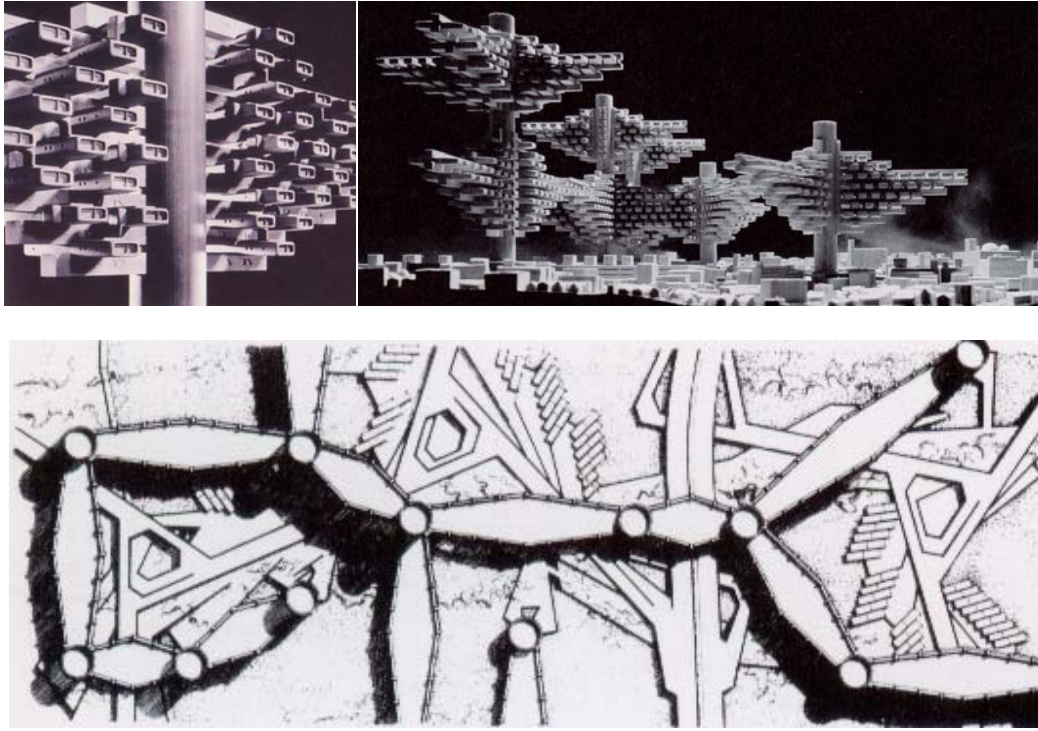
Şekil 2.9 : Habitat 67 (Url-4).

2.2 Toplu Üretilen Modüler Konut Kavramına Ütopik Yaklaşımlar

1950’li yıllarla beraber gelişen standart konut üretimine karşı alternatif bir konut tasarım yaklaşımı da ütopik projeler üzerinden gerçekleştirilmiştir. Özellikle 1970’li yıllarda uluslararası alanda açılan kentsel planlama ve konut bölgelerinin üretimi temalı yarışmalar sonucu elde edilen projelerde, konvansiyonel üretim araçları dışında bir üretim mantığına başvurulduğu gözlemlenebilmektedir. Daha çok A. Isozaki, Kenzo Tange ve Kisho Kurokawa gibi Japon asıllı mimarların çeşitli

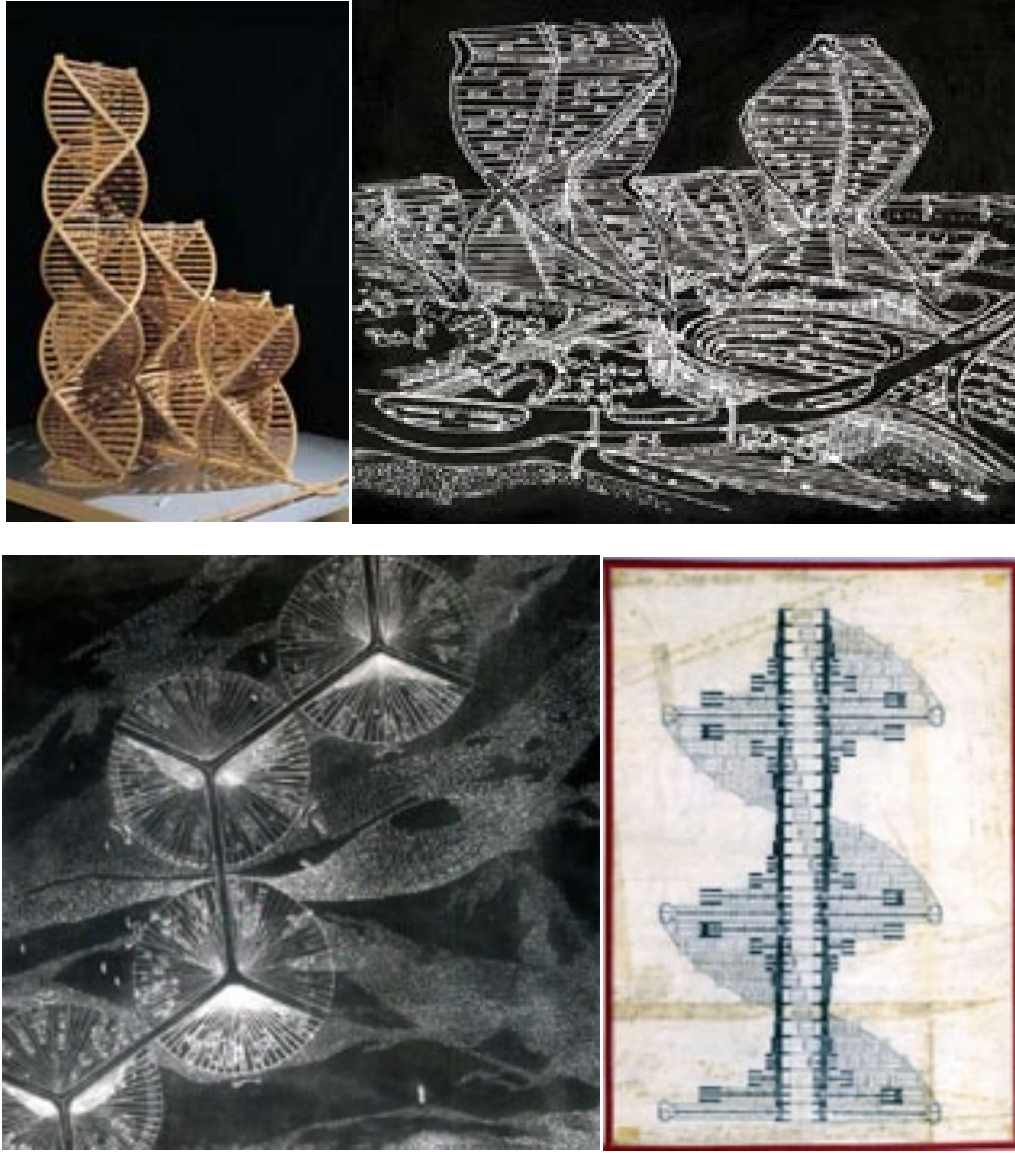
uluslararası yarışmalar için ürettikleri projeler gelecekteki konut planlaması üzerine farklı bir yaklaşım getirmiş olup, irdelenmesi gerekmektedir.

Arota Isozaki'nin farklı zamanlarda yapmış olduğu Clusters ve City in the Air projelerinde (Şekil 2.10) farklı bir konut tasarımına yönelim bulunmaktadır. Tasarım mantığı olarak birbirlerine benzer olan bu projelerde ana bir servis elemanı bulunmakta ve tüm yaşam birimleri bu servis elemanı etrafında üreyen alanlara takılmaktadır. "Cluster in the Air" projesinde "joint core", "City in the Air" projesinde "branches" olarak adlandırılan bu alanlar, konut birimlerine hizmet edecek çekirdek, sirkülasyon, şaftlar ve enerji hatlarını bulundurmaktadır (Kurokawa,1993).



Şekil 2.10 : Cluster-City in the air (Kurokawa,1993).

Konut üretiminde ütöpik sayılabilecek bir diğer yaklaşım ise Kisho Kurokawa'nın Felix City projesinde (Şekil 2.11) gözlemlenebilir. Arota Isozaki'ye benzer şekilde konut birimlerinin yerleşeceği alanlar ve servis gereksinimlerini karşılayacak çekirdekler aynı planlama mantığı ile oluşturulmuştur. Bir helezon veya burgu şeklinde oluşan yapının bu geometrinin sürekli aynı eksende büyümesi özelliği ile çoğaltılabilmektedir (Kurokawa,1993).



Şekil 2.11 : Felix City (Kurokawa,1993).

Üç projenin de ortak noktası olan bir ana çekirdek yapısının bulunması ve konutların bunun etrafında çoğalması betik içerisinde değerlendirilecek bir özelliktir. Bununla beraber yapının gelişirken yeni ikiz yapılara üreterek çoğalmasına benzer organik bir büyüme mantığı da tasarlanacak betik içerisinde ele alınabilir. Fakat boşlukta duran ve giderek çekirdekten uzaklaşan kütleler yerine, çekirdek çevresinde türeyen ve komşuluk ilişkisine sahip olan yaşam birimleri oluşturulmalıdır

2.3 Toplu Üretilen Modüler Konut Kavramına Bilgisayar Destekli Yaklaşımlar

Bilgisayar teknolojilerinin mimari üretim alanına entegre olması, geleneksel hesaplama yöntemleri ile içinden çıkılması zor olan sınıflandırma ve çok boyutlu

optimizasyon problemlerinin daha kolay ve hızlı çözülmesini sağlamıştır. Bunun yanında sonsuz bileşenlere sahip tasarım olasılıklarının biraraya getirilmesini olanaklı kılmıştır. Geleneksel yöntemler ve 60'lı yılların ütöpik yaklaşımlarının aksine bilgisayar destekli tasarım yöntemleri, belirlenen parametrelere göre şekillenen tasarım anlayışını daha mümkün ve anlamlı kılmaktadır. Günümüzde birçok farklı yaklaşım şeması ve programlama altyapısıyla geliştirilen konut projeleri bulunmaktadır. Bu yaklaşımların bazıları, tasarım anlayışı olarak daha sistematize edilmiş kurallar ve kabullenmeler üzerine dayanırken bazıları ise daha futuristik bakış açısı ile problemi yorumlamaktadır. Bu bölümde farklı yöntemler üzerinden bilgisayar destekli tasarım sürecinde gelişen konut projelerine değinilecektir.

Günümüzde konut problematiğı üzerine sıkça fikir yürütmekte olan ofislerden biri MVRDV olarak gösterilebilir. Özellikle modülerite veya çoğul tipolojilerin türetilmesi ile oluşan varyasyonel tasarımlar üzerine çalışmaktadır. Örnek olarak grup “Vertical Village” projesinde (Şekil 2.12) aslen modülerite üzerine gelişen bir proje kurmakla beraber, günümüzde varolan alışlagelmiş yapı kurgusunu ve taşıyıcısı sistem mantığını kırarak bütünsel bir yapı organizasyonuna yönelmeye çalışmıştır (Url-5).



Şekil 2.12 : Vertical Village (Url-5).



Şekil 2.13 : Vertical Village (Url-5).

Basitçe ele alınan modüllerin yani dairelerin birbirleri ile komşuluk ilişkilerini sürekli farklı tutarak bir bütünü oluşturması üzerine gelişen proje, daire varyasyonları veya kullanıcı özelinde bir çeşitlilik sağlamasa da yapısal bütünlük olarak yenilikçi bir anlayışa sahiptir (Şekil 2.13).

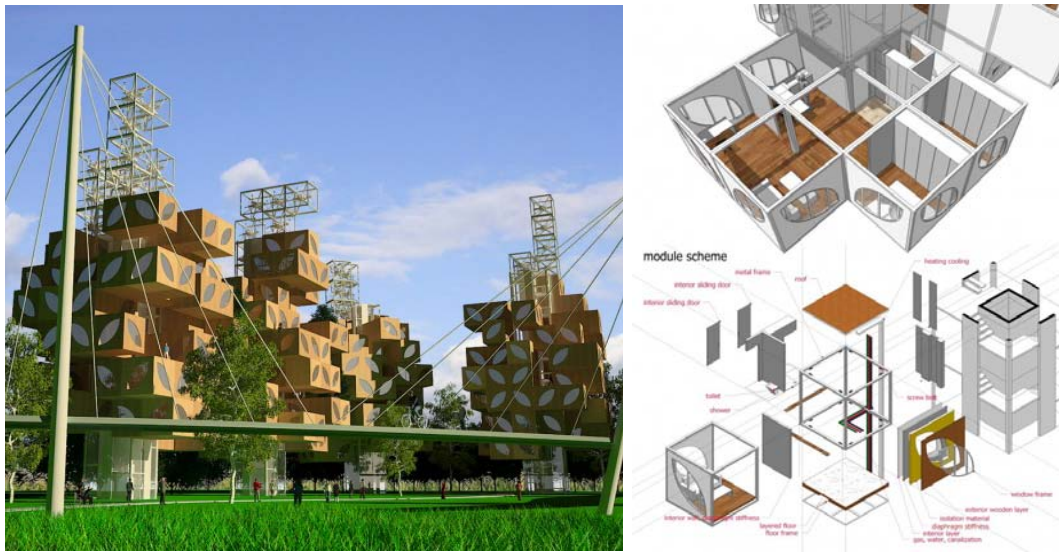
Bilgisayar teknolojileri kullanılarak hazırlanan bir diğer toplu konut projesi Re-Burbia yarışması bünyesinde Adil Azhiyev tarafından hazırlanan T-trees Social Housing konut projesidir. Projede 70’li yıllarda gerçekleştirilen projelerdeki benzerleri bulunan ve “ağaç” yapısına benzer bir altyapı kullanılmıştır. Çekirdek ve asansörü kapsayan bir hacim yaratılmıştır (<http://inhabitat.com>). 3 metre x 3 metre boyutlarındaki küplerin birbirlerine bağlanarak türemesiyle oluşan farklı modüllerin biraraya gelmesi ile daireler oluşmaktadır. Bu modüllerden bir tanesi kesinlikle koridor alanına komşuluk ilişkisi içerisinde olmak zorundadır. Bu sayede dairelere girişler bu alanlardan yapılmaktadır.

Diğer eklenen modüllerde bir önceki modülle komşuluk ilişkisine sahip olmak zorundadır (Şekil 2.14).



Şekil 2.14 : Çekirdek ve modül ilişkisi (Url-6).

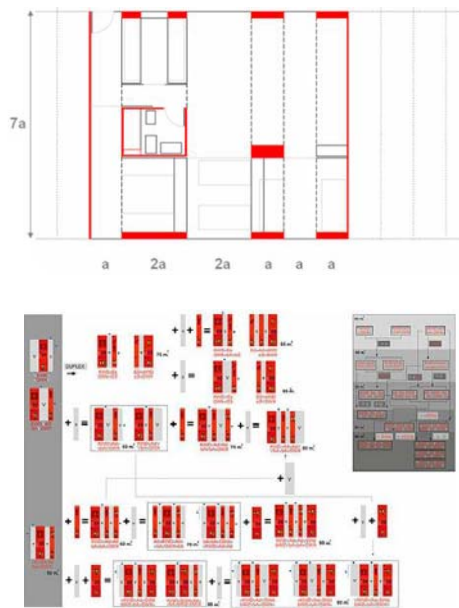
Yaratılan modüller tüm varyasyonlarda farklı şekilde ve sayıda bir araya gelmekte, bazen kapalı bazen ise yarı açık ve açık olacak şekilde alanlar oluşturmaktadırlar. Koridor ile ilişkisi olan ilk modülün koridordaki shaft alanına yakın olması sayesinde tüm ıslak hacim giderlerinin direk shafta bağlanması sağlanmıştır (Şekil 2.15).



Şekil 2.15 : Modül-Kütle (Url-6).

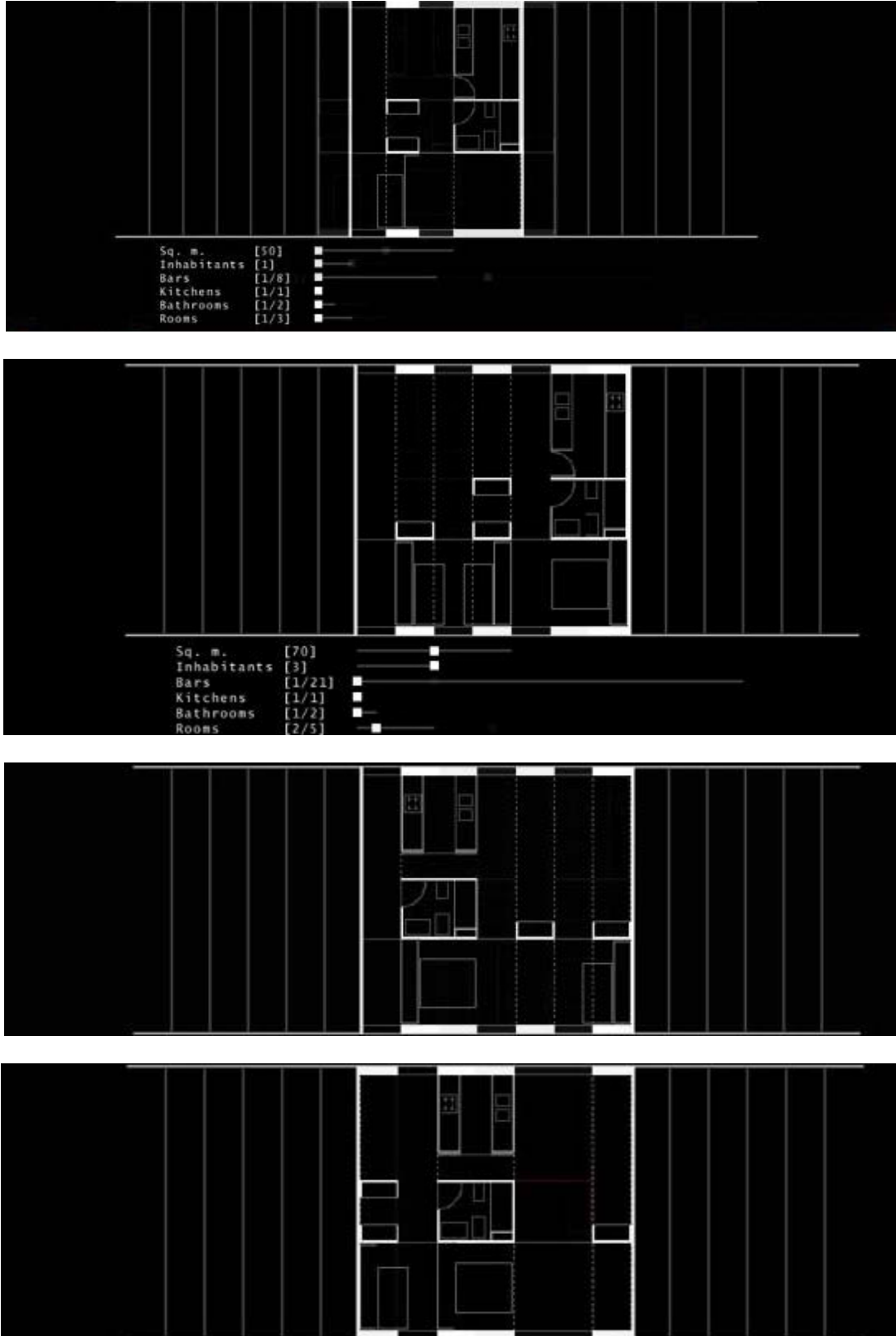
T-trees Social Housing projesindeki en önemli özellik bir çekirdek alanın önceden yaratılması ve ilk modülün daima bu alana komşuluk ilişkisi içerisinde değerlendirilmesidir. Bu sayede plan çözümünde giriş alanı yaratılmıştır. Ayrıca diğer modüllerin sürekli birbirlerini takip ederek çoğalması da daire bütünselliğini sağlamaktadır.

Bilgisayar destekli tasarım anlayışı ile ortaya konulan toplu üretilen konut projelerine bir diğer örnek “Barcode Housing System” projesidir. Bu çalışmanın özelliği bilgisayar destekli yöntemlerle kurgulanan proje mantığı yanında, ortaya çıkan ürünün yapım sürecinin de endüstrileşmiş bir seri üretim mantığına oturtulmaya çalışılmasıdır. Bu çalışmada amaç belirli bir modulerite ve üretim prensipleri üzerine türeyen fakat kullanıcıların veya yaratıcıların dışardan etkisine açık bir üretici (generator) oluşturmak ve bu sayede farklı arsa şartlarında aynı üretim sistematiğine sahip, fakat kullanıcı özelinde farklılaşabilen karakterde bir proje üretmektir (<http://www.barcodehousing.net>). Şekil 2.16 de görülebileceği gibi projede ilk olarak 120 cm genişliği ve 840 cm uzunluğu bir modül boyutu oluşturulmuş ve bu ölçülerin ihtiyaçlara göre değişen daire tipolojilerini elde etmek için sayısal olarak katlanması ile oluşacak bir tasarım ürünü hedeflenmiştir. Belirlenen bu dikdörtgen planlı birimin farklı sayılarda ve bir araya geliş ilişkileriyle her kullanıcının ihtiyaçları çerçevesinde sayısız alternatifte türemesi hedeflenmiştir. “A” sabiti ile tanımlanan birim kendi katları oranında büyüyüp küçülerek veya çeperleri ile bünyesine galeri boşlukları (V) katarak sayısız plan şemasını oluşturmuştur.



Şekil 2.16 : Barcode Housing üretim (Url-7).

Plansal olarak türetmelerdeki ilk kısım belirlenen sabitsel büyüme oranlarında farklı daire tipolojilerinin yaratılması, ikinci kısım ise bunların programa altlık olması için işlenmesidir (Şekil 2.17).



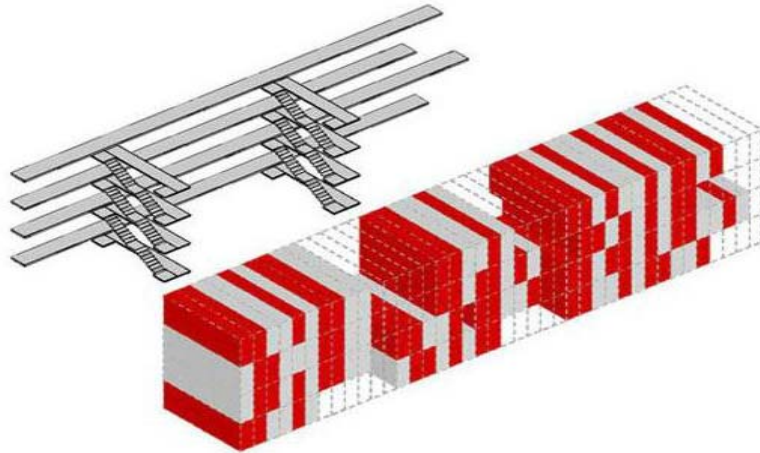
Şekil 2.17 : Kontrol çubukları ile üretilen türetmeler (Url-7).

Program dahilinde yaratılan üretici (generator) kullanıcıya belirlenen kontrol çubukları ile, mutfak, banyo, oda sayıları ve daire büyüklüğüne müdahale etme olanağı sağlamıştır. Türetmeler sadece iki boyut plan düzleminde değil aynı zamanda modüllerin farklı sayılarla üst üste gelmesi ile üçüncü boyutta da oluşturulmuştur. Bununla beraber tasarımcılar modüllerin içerisindeki bölümlenmeleri de esnek ve yer değişebilen ara bölme elemanlarla oluşturarak iç mekanda da çeşitlenmeyi hedeflemişlerdir (Şekil 2.18).



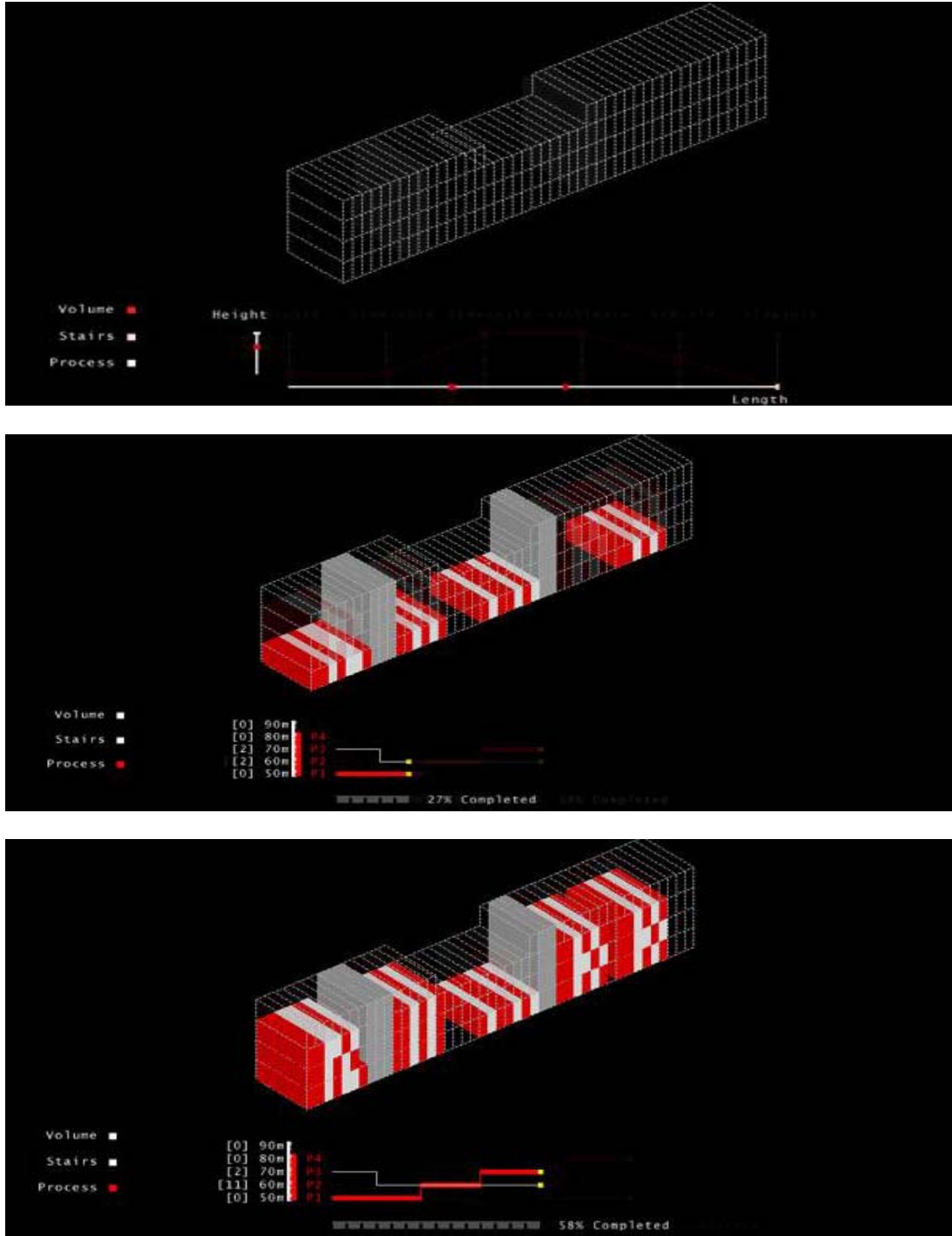
Şekil 2.18 : Ara bölmelerle oluşan farklı iç mekanlar (Url-7).

Proje çerçevesinde tasarlanan yapı tipleri maksimum 4 katlı olarak düşünülmüş ve bununla beraber yatay ve dikey sirkülasyon sistematığı de oluşturulmuştur. Yatay da tek taraflı açık koridor sistemi, dikey de ise belirli mesafelerde tekrarlanarak plana yerleştirilen merdiven alanları oluşturulmuştur (Şekil 2.19).



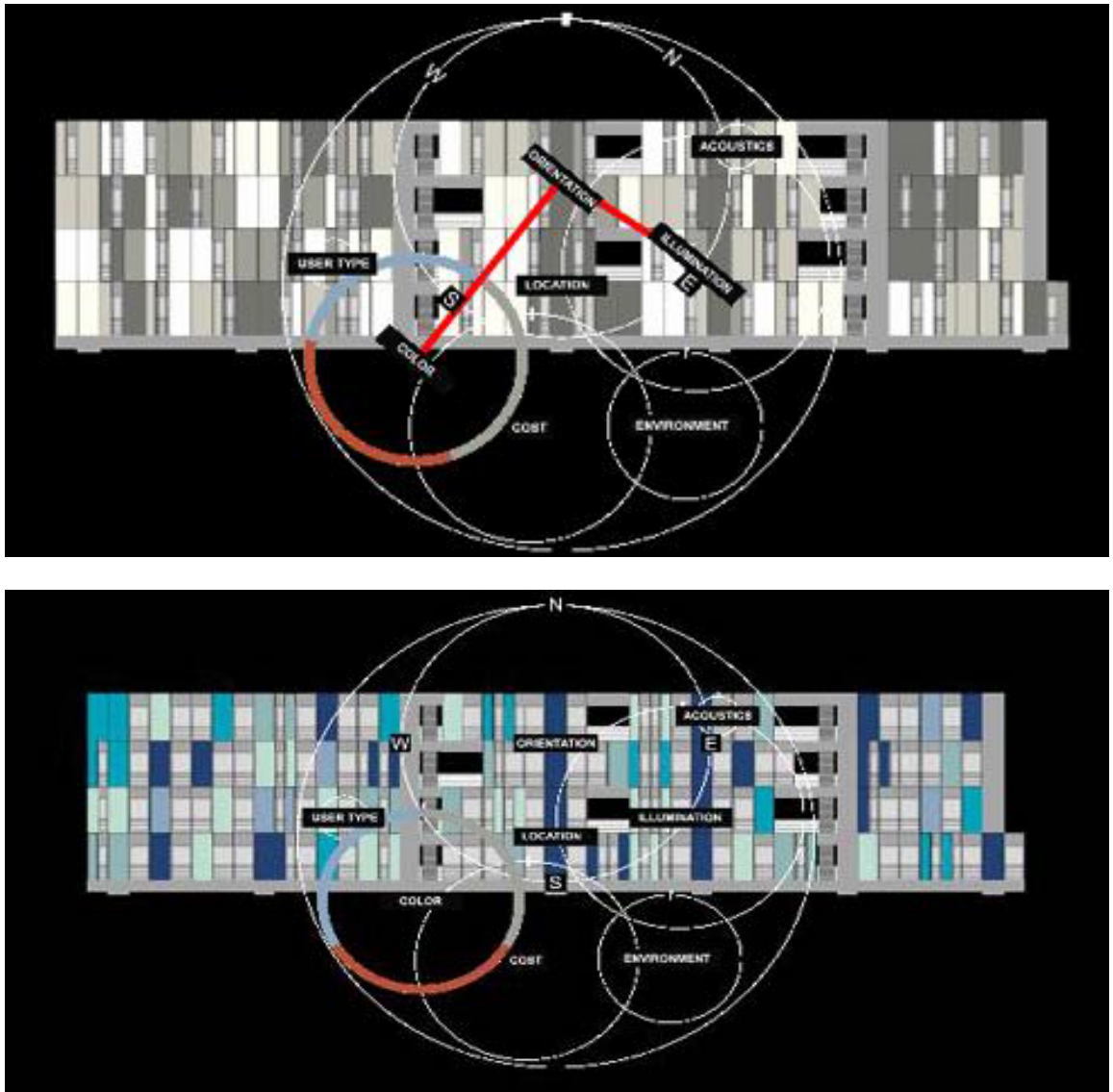
Şekil 2.19 : Yatay ve dikey sirkülasyonlar (Url-7).

Blokların oluřma srecinde ilk olarak belirlenen boyutlardaki alana dikey sirklasyon hacimleri belirli aralıklarla yerleřtirilmekte daha sonra konut birimleri ya otomatik olarak yazılım tarafından ya da kullanıcı tarafından bořluklara konumlandırılmaktadır. Yapının uzunluęu ve ykseklięi yine kontrol ubukları ile ayarlanıp bu ayarlamaya gre de merdiven sayısı ve iki merdiven arası aralık atanmaktadır (řekil 2.20).



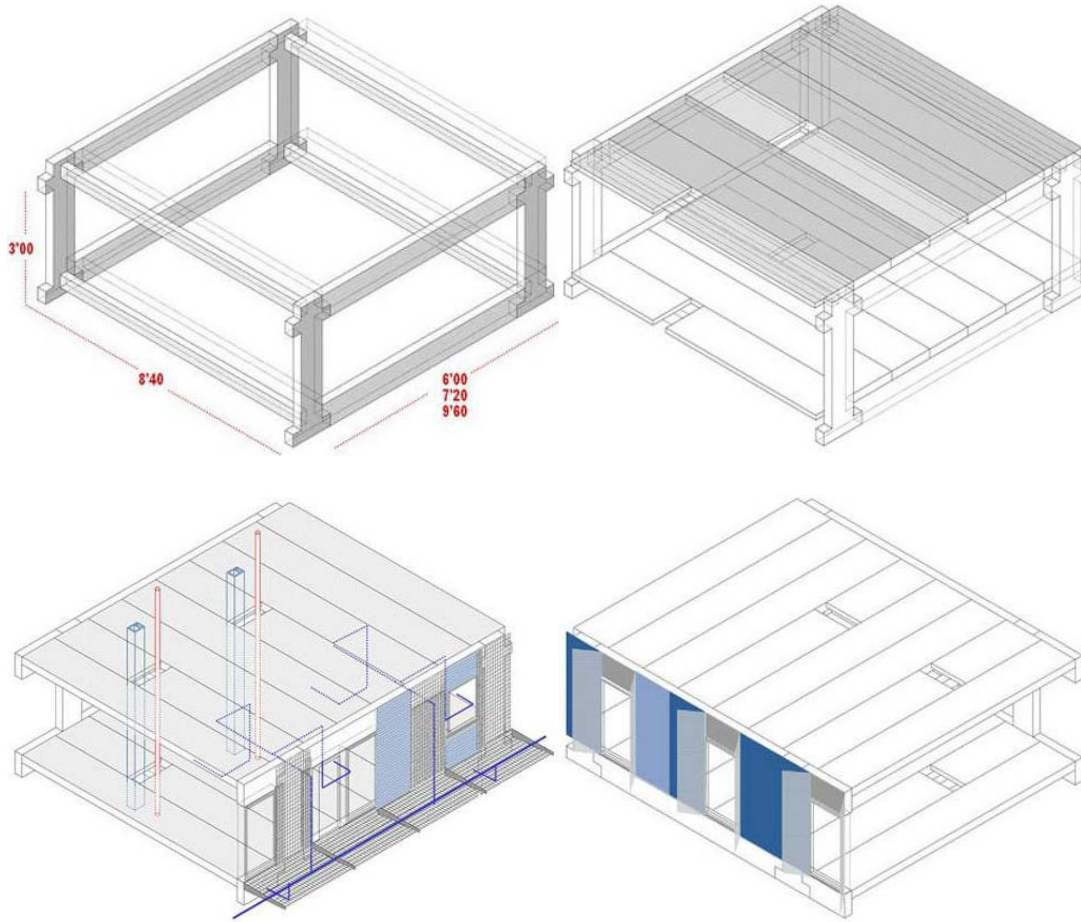
řekil 2.20 : Ktle oluřumu (Url-7).

Tasarım sürecindeki son aşama elde edilen daire yerleşimlerinden sonra cephe kurgusunu oluşturmaktır. Cephe tasarımı projede diğer bileşenlerden bağımsız ele alınmış ve saydam veya opak panellerin belirlenen bir algoritmaya dayalı olarak yerleştirilmesinden oluşmuştur. Cephe tasarımı süreci tamamen kullanıcıya bırakılarak, "ilişkili daireler" olarak adlandırılan diyagramın arayüz üzerinde kullanıcı tarafından yorumlanmasına bağlı olarak yüzlerce farklı alternatif sonuç elde edilebilmesini sağlamıştır. Şekil 2.21'de görüldüğü gibi toplamda 5 parametreden oluşan bu daireleri; oryantasyon, aydınlanma, renk, maliyet ve akustik olarak atanmış ve birbiri ile ilişkilendirilmiştir. Kullanıcı tarafından herhangi birinin değiştirilmesi diğerlerini de etkilemekte ve her seferinde farklı bir cephe kurgusu ortaya çıkmaktadır (Url-7).



Şekil 2.21 : Daireler diyagramı – Cephe kurgusu (Url-7).

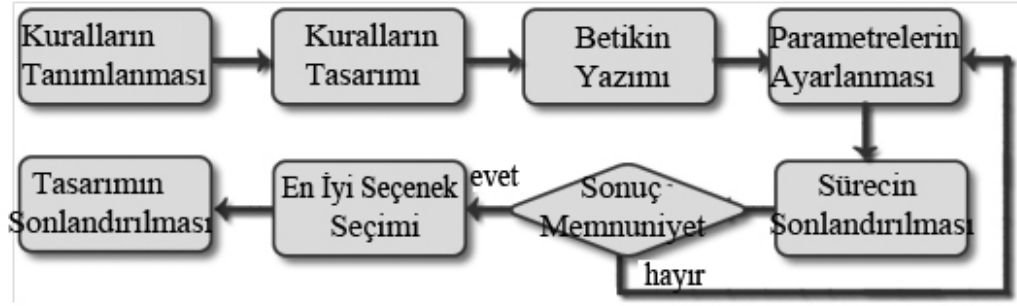
Tasarım sürecinin sonunda yapının nasıl inşa edileceğine dair sistematik çalışmalar da yapılmıştır. Tüm tasarımın bir modül üzerinden şekillenmesi ve sirkülasyon alanlarının basit bir mantıkla yapı içerisine yerleştirilmesi inşa sistematığının de basit bir şekilde ele alınabilmesini sağlamıştır. Prefabrik kiriş kolon ve döşeme elemanları ile belirlenen modüller taşınmakta ve cephe elemanları daha sonra tek cidar şeklinde yapıya yerleştirilmektedir. Şaftlar gibi diğer servis hacimleri de birimlerin koridora bakan cephelerinde kurgulanmaktadır (Şekil 2.22).



Şekil 2.22 : Üretim sistematığı (Url-7).

Barcode projesinde sirkülasyon alanının belirli bi kurala göre belirlenmiş olması ve bu kurala bağlı olarak kendini konumlandırması en önemli noktalardan biridir. T-trees projesi gibi tüm mekanik altyapının ve koridor sisteminin çözümlenmesi ve bir modül boyutundan türetme yönteminin tercih edilmesi tasarlanacak betikte dikkate alınacak özelliklerden birkaçıdır. Ayrıca diğer projelerde cephe kurgusu sadece modüle bağlı olarak oluşurken Barcode projesinde ayrı bir kontrol paneli ile oluşturulabilir olması diğer önemli özelliktir.

Bilgisayar destekli yaratılan diğer bir konut üretim sistemi örneği ise MHAS'tır. Mhas modüler konut birimlerini çeşitli varyasyonlarla bir araya getiren Maya arayüzü ile çalışan bir üretken sistem modelidir (El-Zanfaly, 2009). MHAS Maya betikiğine eklenen bir plug-in olarak çalışan, kullanıcı tarafından belirlenen parametrelerin önceden girildiği, tasarım sürecinin belirlenen kurallar ile oluştuğu ve bu süreçte tasarımcının etkili olmadığı bir modeldir (Şekil 2.23).



Şekil 2.23 : MHAS süreç sistemi (El-Zanfaly, 2009).

Üretken sistem dahilinde kullanıcı ilk önce bütüncül tasarımı oluşturacak modülün boyutlarını belirlemektedir. Daha sonra modüllerin yerleşeceği arazinin boyutları yine kullanıcı tarafından belirlenir. Bununla birlikte arsanın eninin ve genişliğinin başlangıç ve bitiş nokta koordinatları ve modüler yerleşimin başlayacağı ve sonlanacağı nokta koordinatları girilerek tasarım alanı belirlenmiş olunur. Bu aşamadan sonra süreç iki şekilde devam edebilir. Kullanıcı belirlenmiş çerçeve sınırları içerisinde üretimi geliştirebilir veya ikinci şekilde ilk olarak bir sınır içerisinde üretim yapıp daha sonra bu alan üzerine yerleşecek ikinci bir üretim sürecine girebilir (El-Zanfaly, 2009).

Bu aşamadan sonraki diğer aşama ise kullanıcı tarafından “Core” ve “Court” alanlarının tanımlanması kısmıdır. Bu kısımda “Court” tasarım bünyesindeki boşlukları veya açık alanların adet ve boyutlarını tanımlarken, “Core” ise bütün kütlede varolacak servis ve sirkülasyon alanlarına karşılık gelmektedir. Kullanıcı bu alanların sayısını ve boyutlarını girerek tasarım ürünü oluşturmaya başlar. Bununla birlikte alternatifler sekmesine kaç farklı alternatifte ürünün sonuçlandırılmasını istediği bilgisini girerek süreci sonlandırır. Çalışma sonunda iki farklı tasarım elde edilmiştir. Birinci alternatifte 5.5 m. uzunluk 12 m. genişlik ve 3 m. yüksekliğinde 270 adet birimin 25 m. yüksekliğindeki bir alanda üretilmesi örneklenmiştir. Bu üründe ilk aşama üzerinden gidilip ilk aşamada 100 m. x 100 m. lik bir alanda

türetim yapıp daha sonra bu alan üzerindeki yarı oranda bir alanda tasarım sonlanmıştır (El-Zanfaly, 2009)(Şekil 2.24) (Şekil 2.25).

Modular Housing Arrangement System

Unit Parameters

Enter Unit Parameters in Meters

Unit length: 5.0

Unit width: 5.0

Unit Height: 5.0

Units Arrangement in site

Stage 1

Number of Units: 140

Xdir:Start point: 1

Ydir:Start point: 1

Starting Height: 1

Xdir:End point: 50

Ydir:End point: 50

Ending Height: 50

Stage 2

Number of Units: 1

Xdir:Start point: 1

Ydir:Start point: 1

Starting Height: 1

Xdir:End point: 50

Ydir:End point: 50

Ending Height: 50

Court

Parameters

Court Width in meters: 1.0

Court depth in meters: 1.0

Court height in meters: 1.0

Court x-location in meters: 1.0

Court y-location in meters: 1.0

Core

Parameters

Core Width in meters: 1.0

Core depth in meters: 1.0

Core height in meters: 1.0

Core x-location in meters: 1.0

Core y-location in meters: 1.0

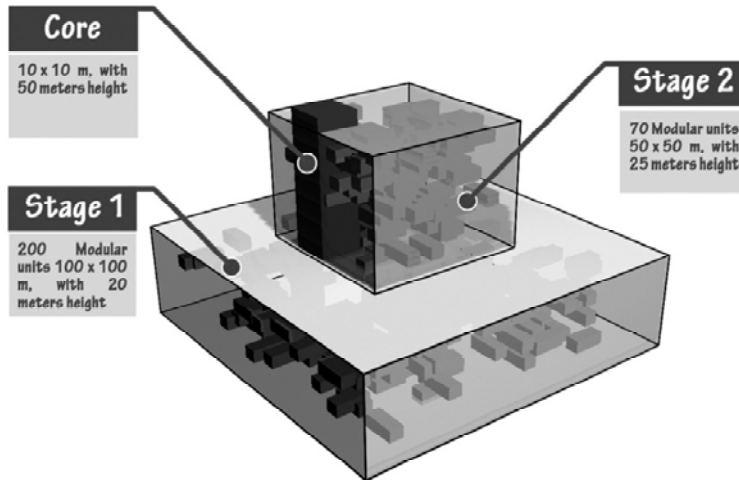
Select a View to render

Alternatives

Number of Alternatives: 1

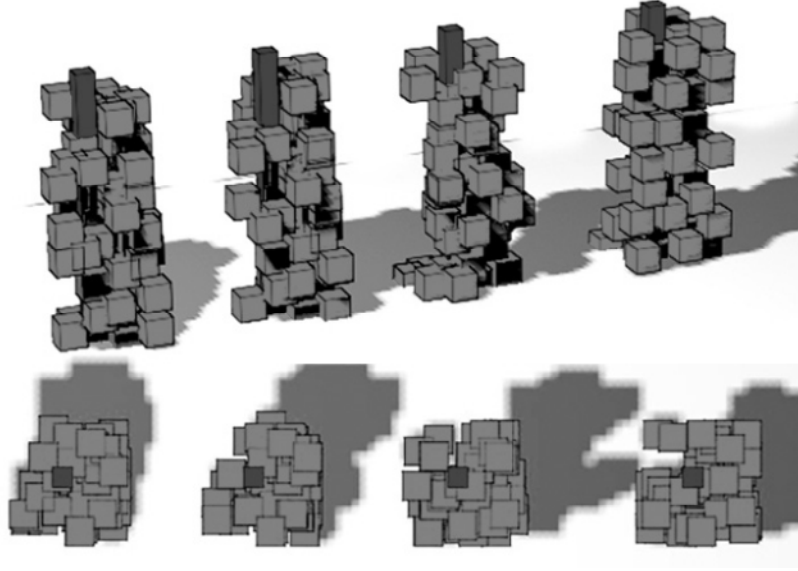
Apply

Şekil 2.24 : MHAS arayüzü (El-Zanfaly, 2009).



Şekil 2.25 : İlk alternatif (El-Zanfaly, 2009).

İkinci alternatif çalışmada ise 5.5 m. uzunluk, 12 m. genişlik ve 3 m. yükseklikteki 70 adet konut modülü 35 m. x 35 m. lik bir alanda rastlantısal olarak bir yüksek yapı oluşturması amaçlanmıştır. Merkezde belirlenen “Core” yani çekirdek etrafına dağılan bu modüller ile tasarım sonlandırılmıştır (Şekil 2.26). Üretken sistemin bir diğer özelliği ise yaratılan ürünün dört farklı cephesini ve bir perspektifini görsel olarak kullanıcıya sağlamasıdır.



Şekil 2.26 : İkinci alternatif (El-Zanfaly, 2009).

Mhas modelinin en önemli özelliği kontrol çubukları veya kullanıcı tarafından girilen bilgilerle tasarım ürününü elde etme olanağıdır. Fakat sürecin gerçek zamanlı olmaması ve oluşan ürünün sadece sürecin en sonunda görülebilmesi tasarımcının her anda sürece dahil olabilmesini engellemektedir. Bunun yanında belirlenen servis alanları ile oluşturulan modüllerin çakışmalarının önüne geçilememiştir. Bu durum da servis alanının ilk olarak bağımsız bir şekilde kurgulanmasının önemini tekrar göstermektedir.

2.4 Değerlendirmeler

Bu bölümde önceki bölümlerde örnekler üzerinden anlatılan yöntemler ele alınacak ve tez bünyesinde oluşturulacak yeni sistem de düşünülerek artıları ve eksileri üzerinden bir anlatım yapılacaktır. Bu sayede bu yöntemlerdeki bazı özelliklerin neden oluşturulacak tasarımda tercih edildiği, bazılarının ise neden tercih edilmediği üzerinde durulacaktır.

Geleneksel yöntemlerde genellikle kullanılan, sistematik bir taşıyıcı ve altyapı mantığının oturtulma fikri, aslında tasarlanan projenin rasyonelliği açısından dikkate alınmalıdır. Bu yaklaşımın üretken sistem betiğini kurgularken kullanılması, konut hücrelerinin bütün içinde ilişkisiz dağılımlarını engelleyecek ve bütün bu birimlerin bir sistematik dahilinde yapıya yerleşmesini sağlayacaktır. Bu yönetime ait örneklerde çeşitlilik sadece ara duvarlar ve cephe boşlukları ile kurgulanmaya çalışılmış ve elde edilen sonuç ürünlerde gerekli görülen çeşitlemeye olanak sağlamamıştır. Alvora Siza'nın Malaqueira konutlarındaki zamana bağlı eklektik çeşitlenmelerin yaratılması bilgisayar ortamında yıllara göre gelişen bir tasarım ürünü yaratılmasında örnek alınabilir. Bununla birlikte bu tür örneklerde genellikle tek tip bir modülerliğin üzerinden türetimin yapılması farklı kütleli hareketleri oluştursa da, aslında her bir modül aynı olup kullanıcısı çerçevesinde değişmiş olmamaktadır.

Ütopik sistemlerde geleneksel yöntemlere benzer olarak sürekli bir servis alanının belirli mesafelerle tekrar edecek şekilde yapının omurgasını oluşturma fikri ele alınmalıdır. Merkezde bir çekirdek alanının bulunması ve tüm modüllerin bunun etrafında şekillenmesi üretim sistemi açısından gerekli olmakla beraber yine sadece tek bir modül üzerinden üreyen bir organizmanın kurulmuş olması hücre zenginliğini en aza indirmektedir.

Bilgisayar destekli tasarım yöntemlerinde özellikle kullanılan kontrol çubuğu mantığı ve kullanıcıya parametreleri belirleme şansı verilmesi bu yöntemlerdeki en önemli özelliktir. Fakat üretken sistemlerdeki ürünü sadece sonuç bölümünde elde etme mantığı yerine, gerçek zamanlı değişimlerin kullanıcı tarafından izlenebilmesi gerekmektedir. Böylece kullanıcı veya tasarımcı kendi yönelimleri doğrultusunda süreci sonlandırabilecektir. Bununla birlikte yine ebatları belirlenmiş tek bir modülden yapılan türetmeler yerine önceden belirlenen çeşitli modüllerin bir araya gelebilmesini sağlamak ve bu birlikteliklerden çıkan varyasyonları ele almak projeyi zenginleştirecektir. Bu süreçte karşılaşılabilecek en önemli sorun servis ve sirkülasyon sistemlerinin rasyonel bir şekilde üretken sistem bünyesinde tanıtılabilmesidir. Bu aşamada bu alanlar ne tasarımdaki yaratıcılığı kısıtlayacak şekilde geleneksel çözümlere yönelmeli ne de yapı birimlerinin mantıksallık dışında üretilmesine olanak vermelidir.

Tanımlanacak betik ve tasarlanacak üretken sistem çerçevesinde bu yöntemlerin olumlu ve olumsuz tarafları irdelenecek ve belirli ölçülerde sistem içerisinde kullanılacaktır.

3. MODÜLER KONUT TASARIMI İÇİN GELİŞTİRİLMİŞ PARAMETRİK TASARIM TABANLI MODEL

Günümüz mimarisinde, 2. bölümde de üzerinde durulduğu gibi bilgisayar destekli tasarım çalışmaları konut planlamasında önemli bir yer tutmaktadır. Son yıllarda, bilgisayar teknolojileri ve mimarlık alanının daha yoğun etkileşimi ile parametrelere bağlı girdiler aynı anda değerlendirilebilmektedir.

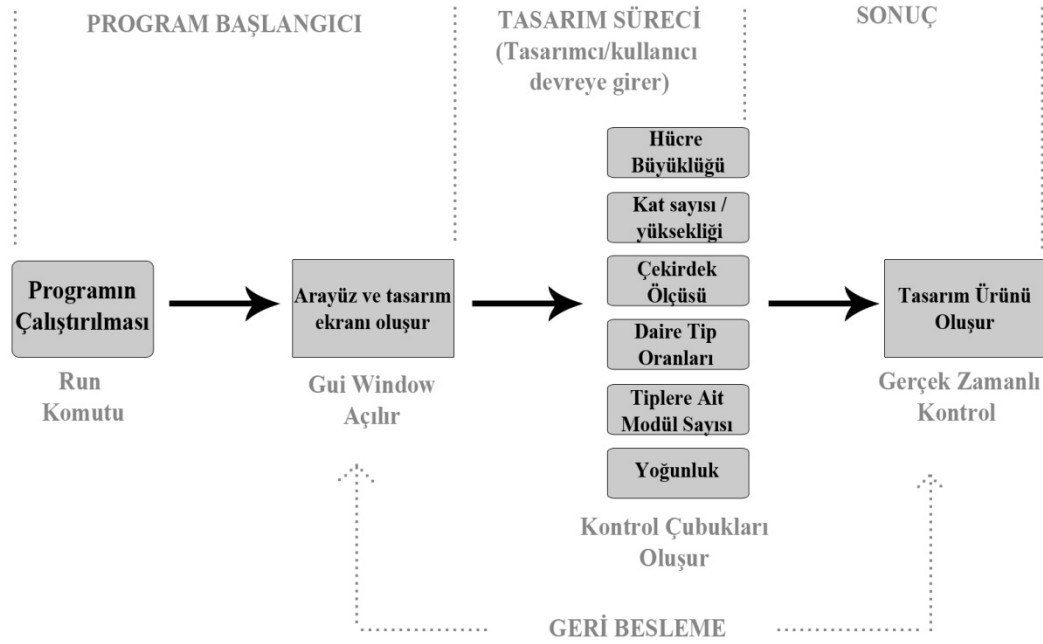
Tez kapsamında, kullanıcı parametrelerine bağlı olarak gelişen ve aynı anda birçok türevde tasarım ürünü ortaya koyabilen bir model üzerinde çalışılmıştır. Yazılan algoritmalar ile belirli bir sistematığe sahip toplu konut projeleri oluşturulabilmektedir. Geleneksel yöntemler ile sadece bir tip kullanıcı için, belirli ürünler elde edilebilirken yaratılan modelle aynı anda birçok parametreye bağlı çoğul tasarım ürünleri elde edilebilmektedir.

3.1 Geliştirilen Modelin Amacı ve Tanıtımı

Tez çalışması kapsamında amaç farklı kullanıcı tipolojileri için aynı anda birçok alternatifli yaşam üniteleri oluşturan bir üreteç tasarlamaktır. Oluşturulmak istenen algoritmanın tasarımcı tarafından “gerçek zamanlı kontrol edilebilir” olması aranan en önemli özelliklerden biridir. Bununla beraber, üreteç aracılığıyla oluşturulan kütlenin mimari ve mühendislik kurallarına cevap veren ve sorunlara karşı rasyonel çözümler üreten bir yapıda olması amaçlanmıştır. Model algoritmasının daha sonraları eklenecek yeni parametrelere açık olması ve istenildiği an yeni parametreler eklenerek parametre kontrol çubukları ile arayüzün geliştirilebilmesinin mümkün olması için çalışılmıştır. Üretilcek programın diğer programlar ile etkileşimli bir yapıda olması ve istenildiği zaman modelin diğer programlara aktarılabilir sunum veya tasarım sürecinin diğer ortamlarda devam ettirilebilmesinin önü açılmıştır.

Program tüm alt bileşenlerini aynı anda çalıştırıp, ürünleri aynı zamanda oluştursa da, ilk olarak yapının servis alanına hizmet verecek alanın yaratılması ile tasarıma başlandığı söylenebilir. Bu aşamayı daha sonra konut birimlerinin sayısal

değerlerinin girilmesi izlemektedir. Yapıda bulunacak kat sayısının ve katlara ait metrik bilgilerin belirlenmesi ile süreç devam edilmekte ve diğer tüm parametreler bu aşamalardan sonra istenildiği şekilde çeşitlendirilebilmektedir (Şekil 3.1).



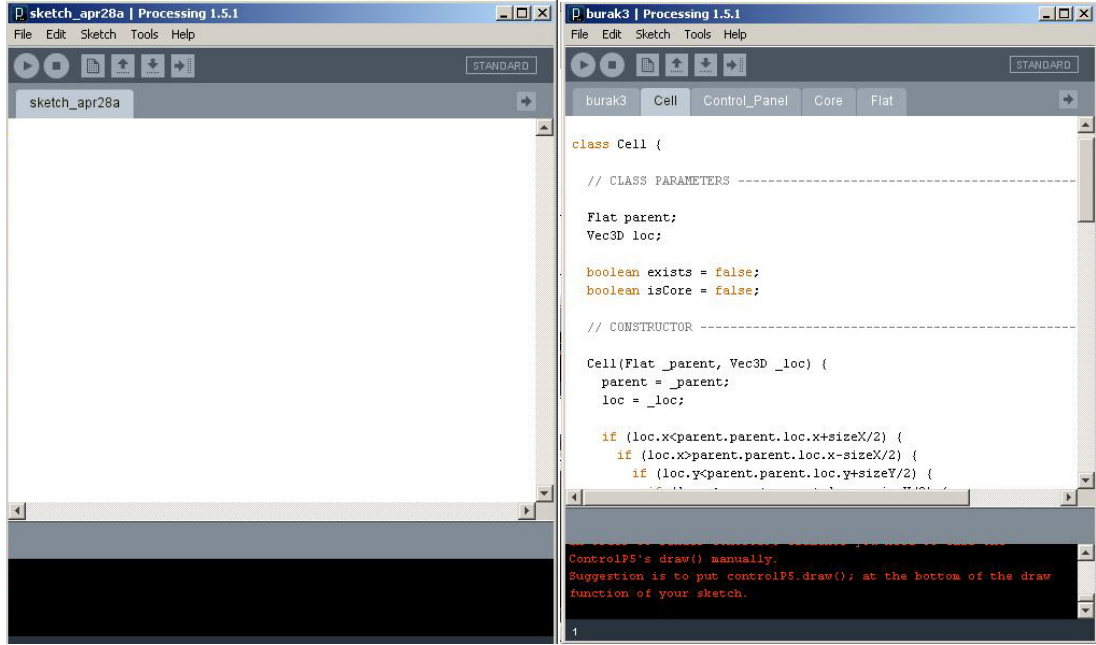
Şekil 3.1 : Geliştirilen modelin algoritması.

Modelin girilen parametre ve değerlere bağlı olarak ürettiği tüm daire tipolojilerinden sonra birimlerin ayrı ayrı ele alınıp plansal çözümlerinin oluşturulması daha sonraki işlem basamağını oluşturmaktadır. Mimari çizim programları kullanılarak istenilen daire tipleri ve çekirdek alanı plan bazında çözümlenmektedir. En son olarak da model başka bir program diline çevrilerek cephe kurgusu oluşturulmaktadır.

3.2 Modelin Algoritması

Kullanıcı gereksinimlerine bağlı parametrik tabanlı modelin yazıldığı algoritma, “PROCESSING” yazılım geliştirme aracı kullanılarak oluşturulmuştur. Processing, 2001 yılında MIT bünyesinde oluşturulmaya başlanan daha sonraları ise Ben Fry ve Casey Raes tarafından geliştirilen bir programdır. Mimarlık ile beraber birçok dalda kullanılabilen programın en önemli özelliği açık kaynaklı bir altyapısının olması ve bağımsız kullanıcılar tarafından sürekli geliştirilebilir olmasıdır. Bu açık kaynaklı altyapı sayesinde “Library” (kütüphane) adı verilen ve daha önceden başka kullanıcılar tarafından oluşturulan bağımsız çalışma kodlarının istenildiği şekilde

programa adapte edilebilmesidir. Bununla beraber program çok basit bir arayüze sahiptir (Şekil 3.2). Bu sayede yazılacak betik ile kullanıcı arasına herhangi ikinci bir arayüz girmemekte ve yazılan kod aynı anda kolayca çalıştırılarak sonuç görülebilmektedir.



Şekil 3.2 : Processing arayüzü-sekmeler.

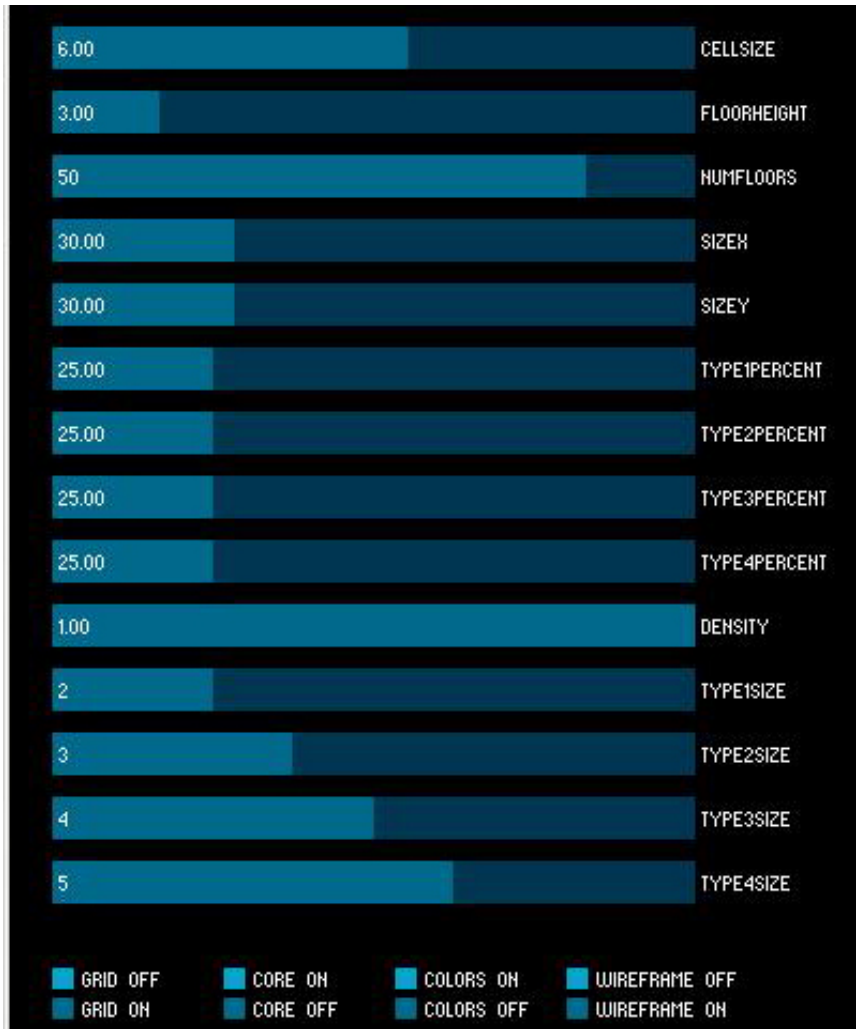
Hâlihazırda gerçek zamanlı çalışabilme özelliğine sahip olmayan programa daha önce hazırlanan kütüphaneler eklenerek bu özellik atanmış olur. Bu kütüphaneler “TAB” (Sekme) adı verilen bölümler halinde programa eklenebilmektedir. Aynı ayrı düzenlenen fakat aynı betik altında çalışan bu sekmeler sayesinde karışık görünen düzensiz betik dizileri yerine, kümeleşebilen betikler oluşturulabilmektedir (Şekil 3.2).

3.3 Modelin Arayüzü

Model arayüzü algoritmaların yazıldığı betik kısmı ve betiğin çalıştırılması ile ortaya çıkan kontrol paneli ile görselleştirme ekranından oluşmaktadır. Betik kısmı, tüm algoritmayı kapsar ve kendi içinde sekmelere ayrılır. Bu sekmelerin her biri projede bulunan çekirdek, katlar, modüller ve tipolojiler olarak oluşturulmuştur. Modelin arayüzündeki iki en önemli özellik kütüphaneler ve kontrol çubukları olarak sınıflandırılabilir.

Kontrol çubukları

Modelin, betiğini hazırlamış tasarımcısı dışında, kullanıcılar tarafından da rahatlıkla kullanılabilmesi için kontrol çubukları oluşturulmuştur. Bu kontrol çubukları sayesinde tasarımcı, betik sisteminin karmaşıklığından kurtulup basit bir arayüz ve kontrol çubukları ile istediği parametreleri sisteme girebilmektedir. Böylece herkes için rahatlıkça kontrol edilebilecek bir tasarım arayüzü oluşturulmuştur. Bununla birlikte kontrol çubukları sayesinde değişimler gerçek zamanlı olarak model tarafından algılanabilmekte ve betikin durdurulup tekrar çalıştırılmasına gerek kalmamaktadır. Kontrol çubukları ile kullanıcı veya tasarımcı an ve an tasarımın değişimini gözlemleyerek son ürüne kolayca karar verebilmektedir (Şekil 3.3).



Şekil 3.3 : Kontrol çubukları.

Kontrol çubuklarının diğer bir özelliği ise kolayca arayüze bütünleştirilebilmeleridir. Betikte olan tüm parametreler rahatça ve aynı sistematikte kontrol çubukları şekline döndürülebilmektedir. Siyah arka planlı bir arayüz üzerinde gelen kontrol çubukları

bu sayede basitçe kullanılabilir. Kontrol çubukları ile ayarlanan değerler sırasıyla hücre boyutları, kat yüksekliği, kat sayısı, çekirdek alanının x ve y koordinatlarındaki ebatları, daire tiplerinin oransal dağılımları, projedeki toplam yoğunluk katsayısı ve tiplerin kaç adet modülden oluştuğudur. Hücre boyutları minimum 0, maksimum 10 metre arasında tercih edilebilir. Daire yükseklikleri 2 metre ile 8 metre arasındaki değerlerde atanabilir. Projedeki toplam kat sayısı 0 ile 60 kat arası değiştirilebilirken çekirdek alanının büyüklüğü de her iki ekseninde 10 metre ile 80 metre arasındaki değerlerde oluşturulabilir. Tüm daire tiplerinin yoğunlukları 1' den 100'e kadar olan bir değerde girilebilir fakat kullanıcının girdiği dört ayrı değer toplamının 100 olması gerekmektedir. Bu sayede %100'lük bir dağılım sağlanabilir. Yoğunluk katsayısı 0 ile 1 arasındaki değerlerdedir ve yoğunluğu göre çekirdek etrafına gelen daire sayıları azalır veya artar. Son parametreler olan daire modül sayıları ise her tip için 0 ile arasında seçilebilir.

Kütüphane sistemi

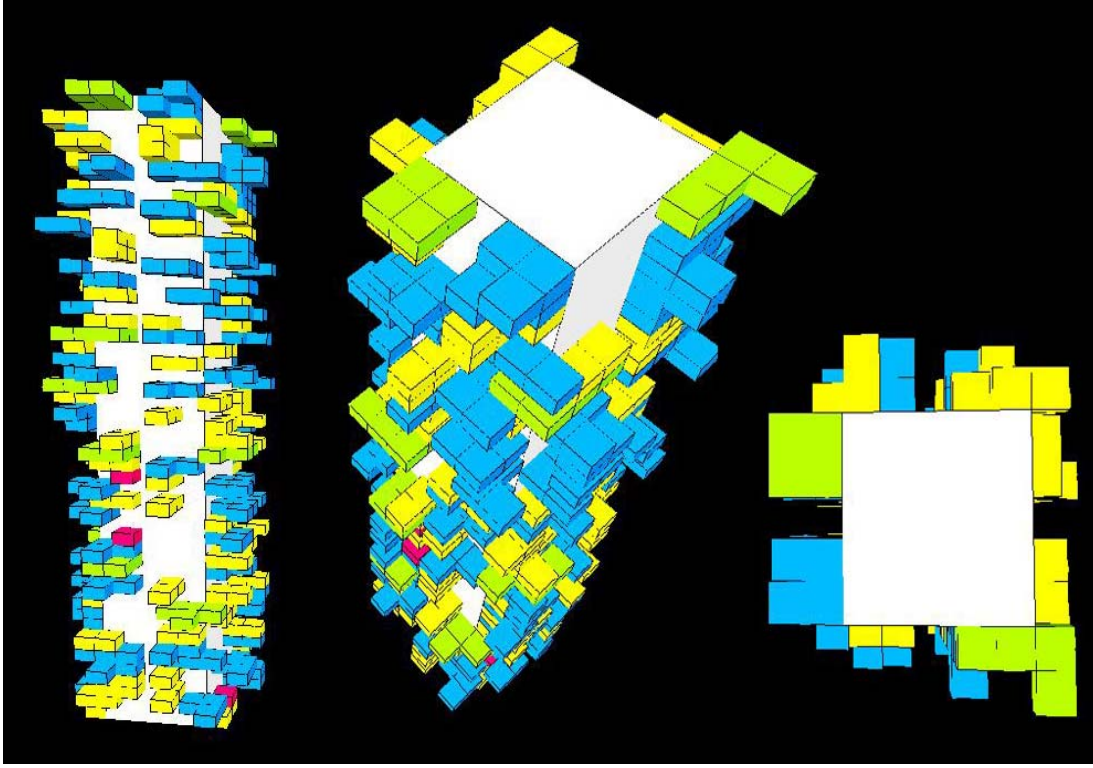
Kütüphane sistemleri açık kaynaklı gelişim özelliğine sahip ve internet üzerinden birçok kullanıcı tarafından eklemeler yapılabilen yazılım programlarının en önemli özelliklerinden biridir. Bu tip programlar sayesinde dünyanın her tarafından kullanıcılar yeni geliştirdikleri program alt bileşenlerini veya projeleri diğer kullanıcılar ile paylaşabilmektedirler. Yazılım programları yanında bazı BIM programlarında da benzer özelliklere rastlanabilmektedir.

Model bünyesinde kullanılan bazı özellikler bu kütüphanelerin geliştirilmesi ile oluşturulmuştur. Model bünyesinde toplamda 3 adet kütüphaneden yararlanılmıştır. Örnekleme gerekirse, kontrol çubuklarının oluşturulmasında kullanılan “controlP5” kütüphanesi model çerçevesinde geliştirilerek modele dahil edilmiştir. Bu sayede kontrol çubukları kolayca oluşturulabilmiştir.

Model kullanılan “toxi.geom” kütüphanesi ile ise iki ve üç boyutlu vektörlerin oluşturulması, yüzeylerin görselleştirilebilmesi ve diğer poligonal alanların görselleştirilebilmesi sağlanmıştır.

Modelde kullanılan son kütüphane ise, modelin yarattığı yapı önerisi etrafında 3 boyutlu olarak kamera sayesinde hareket edebilme ve tasarım ürününü farklı açılardan izleyebilme özelliğini atayan kütüphanedir. PeasyCam adı verilen bu kütüphane sayesinde tasarım ürünün farklı açılardan incelenebilmektedir. Böylece

daha etkili bir mimari proje sürecine sahip olunmakta ve istenildiği zaman plan, görünüş veya yapıya ait farklı açılardan perspektifler elde edilebilmektedir. (Şekil 3.4).

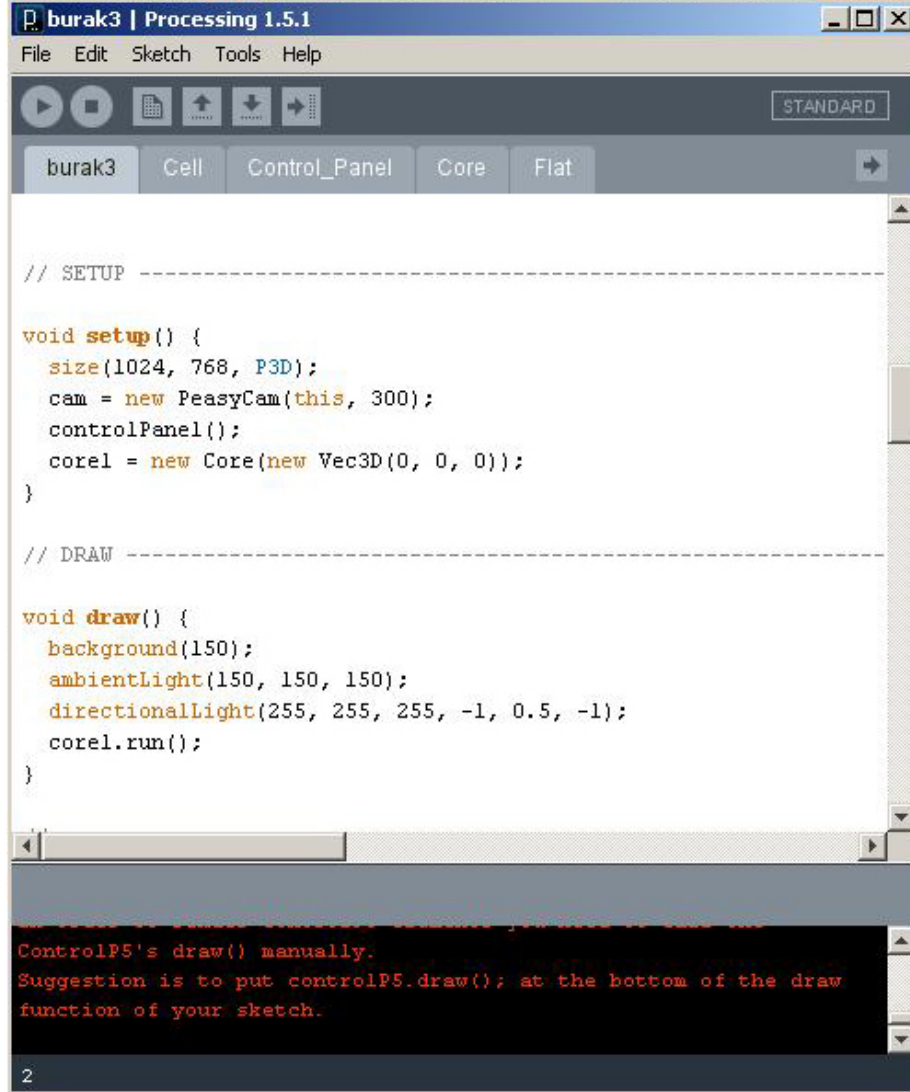


Şekil 3.4 : Farklı bakış açıları.

3.4 Modelin Çalışma Aşamaları

Yazılan modelde ilk olarak üzerinde çalışılacak projenin tasarım alanı özellikleri belirlenmiştir. Kullanılan modelde yaratılan ilk sekmede “setup” ve “draw” başlıkları altında çalışma arayüzü özellikleri girilen parametrelerle belirlenmiştir. Tasarım ürünün görsel olarak izlenebileceği alan “size” parametresinde görülebilmektedir. Bununla beraber arka plan rengi, görselleştirme esnasında kullanılacak ışık ayarları yine bu bölüm içerisinde oluşturulmuştur. Tasarım ürünün 3 boyutlu olarak izlenebilmesi için kullanılan “PeasyCam” kütüphanesi, kontrol çubuklarının oluşturulmasında kullanılan “ControlP5” ve geometrilerin tanımlanmasında kullanılan “toxi.geom” kütüphaneleri de bu bölümde modele tanıtılmıştır. Tanıtılan kütüphaneler yanında modelin arayüzünde kullanılacak olan ışık değerleri yine bu bölümde tanımlanmıştır. Bütün belirlenen bu özellikler kullanıcı tarafından

değiştirilebilir olup, farklı ebatlarda ve özelliklerde çalışma alanları oluşturulabilmektedir (Şekil 3.5).



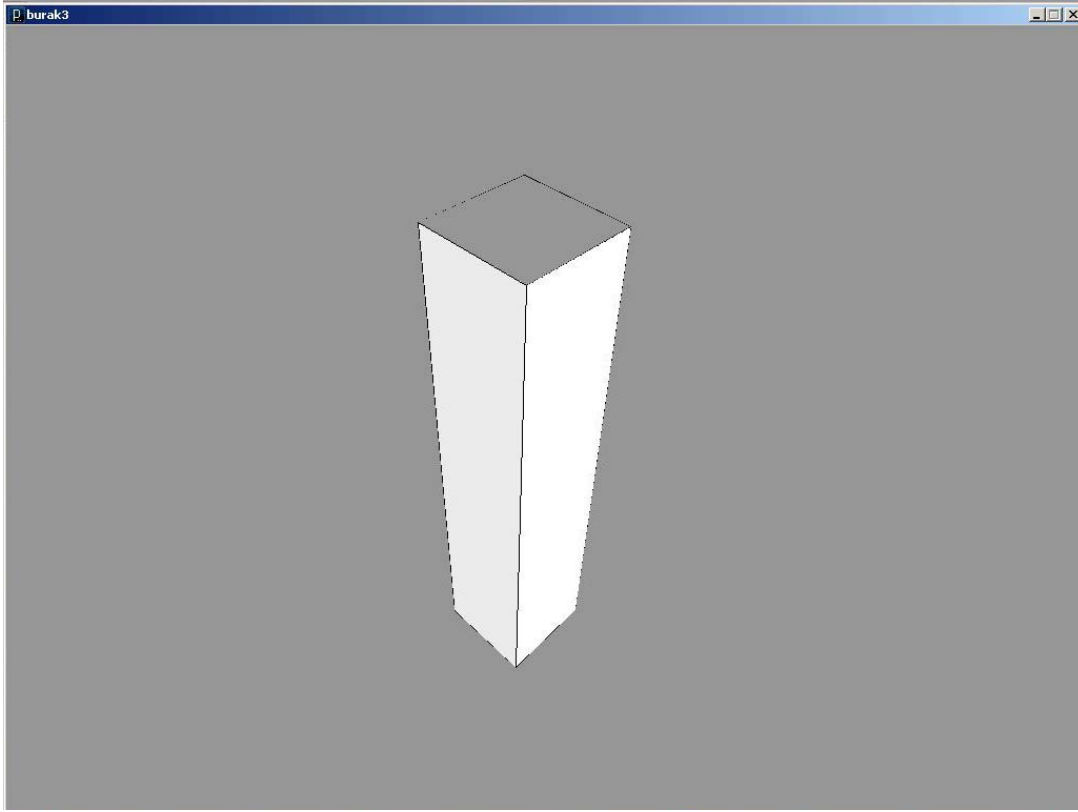
Şekil 3.5 : Processing “Setup” ve “Draw” bölümleri.

Çekirdek ve servis alanı

Hazırlanan modelde tasarımın oluşturulacağı alan özellikleri belirlendikten sonra modelin oluşturulmasına geçilmiştir. Betiğin çalıştırılması ile tüm yapı elemanları aynı anda arayüzde belirse de ilk olarak taşıyıcının oluşturulması ile modele başlandığı söylenebilir.

Çekirdek tasarlanırken ikinci bölümde ele alınan örnekler üzerine yapılan değerlendirmeler dikkate alınmıştır. Ütopik yaklaşımlarla üretilen “City in the Air” ve “Felix City” ile bilgisayar destekli tasarım yaklaşımlardan “Barcode” projesinin sahip olduğu çekirdek yaklaşımına yakın bir konsept üzerine gidilmiştir. Çekirdek,

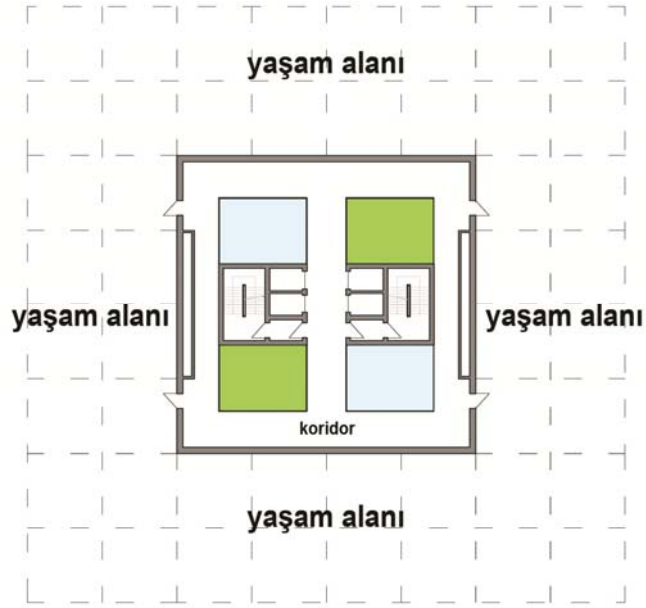
merkezde yer alan ve yaşam birimlerini çevresine alacak bir şekilde düşünülmüştür. Çekirdek ve diğer tüm servis alanlarının merkezde düşünülmesi ile hem daireler ve sirkülasyon alanı arası komşuluk ilişkileri rasyonel bir şekilde oluşturulabilmiş hem de betiğin daire alanları ile çekirdek alanlarının ayrımını kesin olarak yapabilmesi sağlanmıştır. Bu sayede dikey ve yatay izdüşümde daire alanları ile çekirdek ve servis alanları birbirinden tamamen ayrılmış ve kesişmeleri olanaksız hale getirilmiştir (Şekil 3.6).



Şekil 3.6 : Çekirdek alanının oluşumu.

Çekirdek alanının büyüklüğü daha önceki bölümlerde belirtildiği gibi, yazılan betikte kullanıcı tarafından istenildiği şekilde belirlenebilmektedir. Buna rağmen en uygun çekirdek alanı ölçüleri 20 m. x 20 m. olarak düşünülmüştür. Bu ölçülendirme yapılırken tasarlanılacak olan tek bir modülün ebatları dikkate alınmıştır. Genel olarak çekirdek alanının ebatlarının tek bir modülün kenar ölçüsünün katı olacak şekilde oluşturulması, planlamanın rasyonelliği açısından önemlidir. Optimum koşulda çekirdek alanının bir kenarının ebadı bir modülün kenar ebadının dört katı şeklinde düşünülebilir. Yani modül büyüklüğünün 5 metre çekirdek kenar ebadının da 20 m. olduğu durum tasarım açısından optimum değerler olarak alınır.

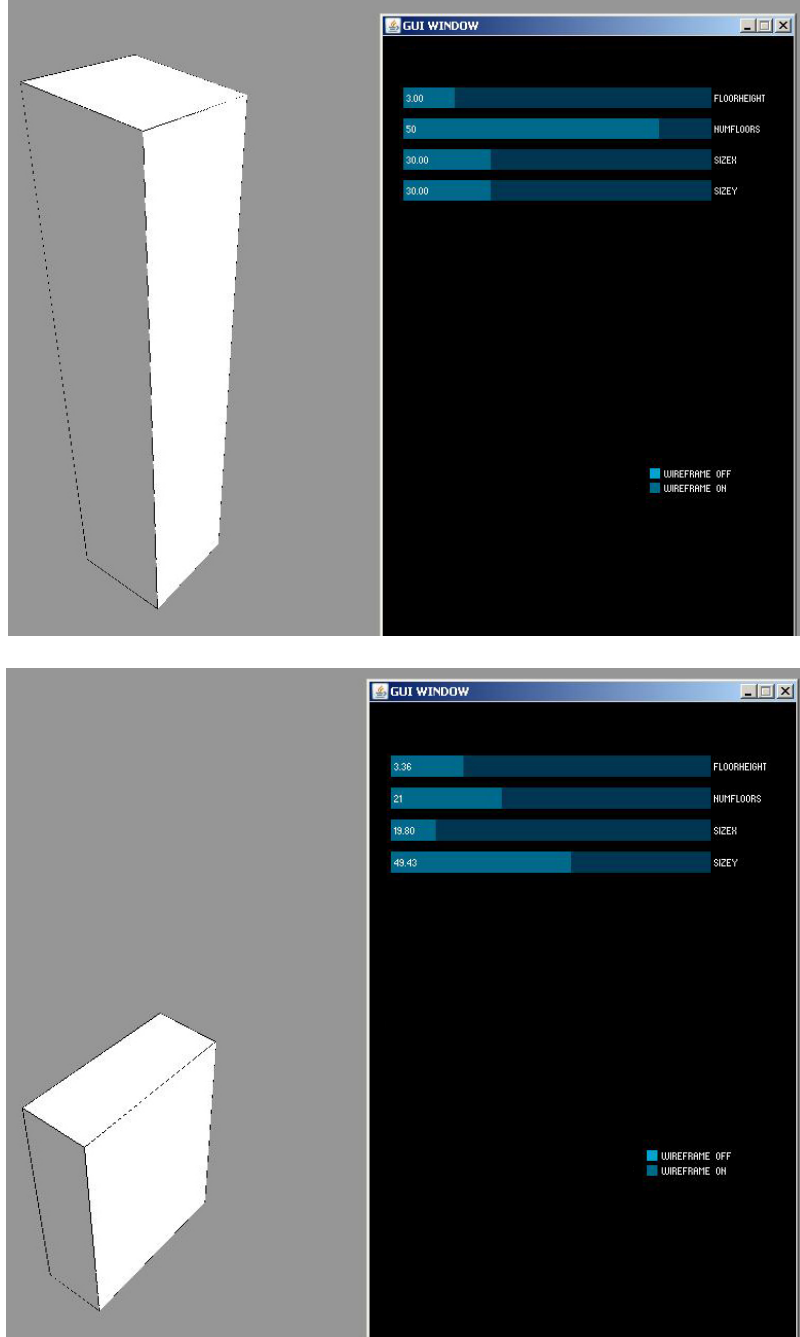
Bu çekirdek alanı genel anlamda plan bazında; galeri boşlukları, kat merdiveni, yangın merdiveni, asansör, elektrik ve mekanik şaftlar ve yatay dolaşım alanı olan kat koridorlarını bünyesinde bulundurmaktadır (Şekil 3.7).



Şekil 3.7 : Çekirdek alanı plan ve modeli.

Model içerisinde çekirdek yapısı ile ilgili olarak genişlik-uzunluk, kat yüksekliği ve kat sayısı belirlenen parametrelerdir. Bu parametreler hakkında daha detaylı anlatım yapılacak kontrol çubukları ile kullanıcı tarafından kontrol edilebilmektedir. Genişlik ve uzunluk gibi metrik değerler diğer yapı elemanlarından bağımsız bir şekilde değiştirilebilirken, kat yüksekliği ve sayısı parametreleri modüllere ait parametreler

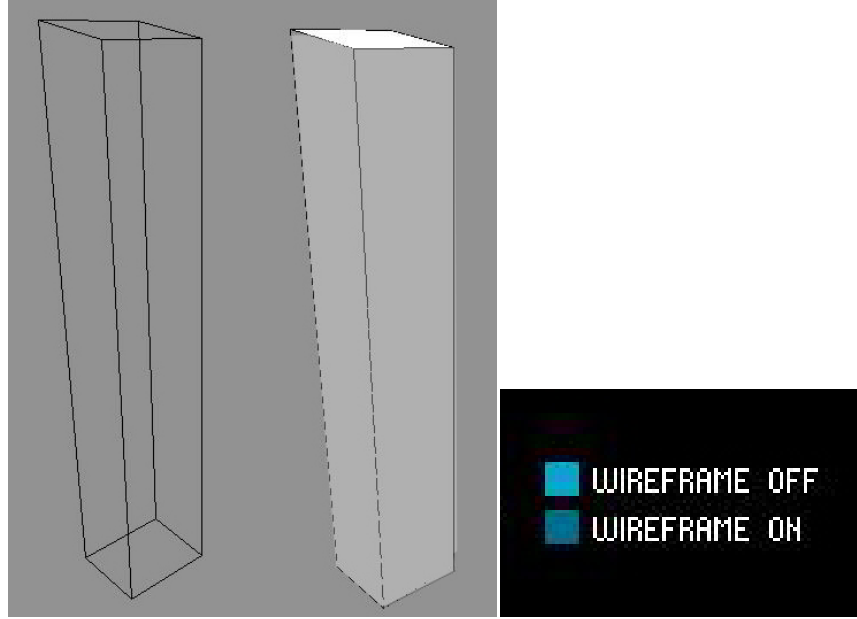
ile ilişkilendirilmiştir. Bu sayede çekirdekte yapılan herhangi bir kat yüksekliği veya kat adedi artışı paralel olarak modüllerde ve daire birimlerinde de oluşmakta, iki birim arasında bir kural ilişkisi kurulmaktadır. Şekil 3.8’ da karşılaştırmalı olarak farklı parametrelerle elde edilen çekirdek alanı görülebilmektedir.



Şekil 3.8 : Karşılaştırmalı çekirdek alanları.

Belitilen bu parametreler ile birlikte, çekirdeğin tasarımcı tarafından görsel olarak yüzeyler veya tel örgü halinde etkinleştirilebilmesi sağlanmıştır. Arayüzde kontrol

panellerinin sağ alt kısmında bulunan tel örgü açıp kapama (wireframe on/off) modları ile çekirde alanının nasıl görselleştirileceği seçilebilir (Şekil 3.9).



Şekil 3.9 : Yüzeysel ve tel örgü çekirdek.

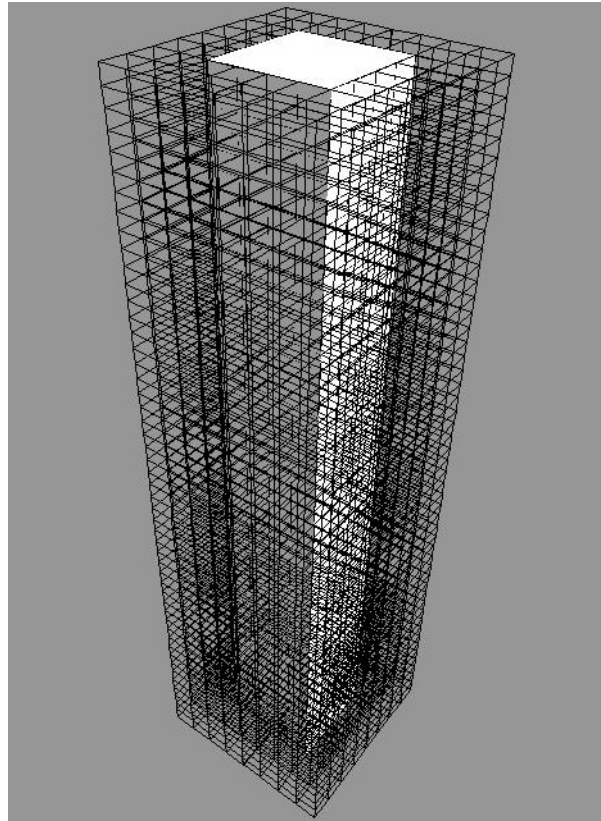
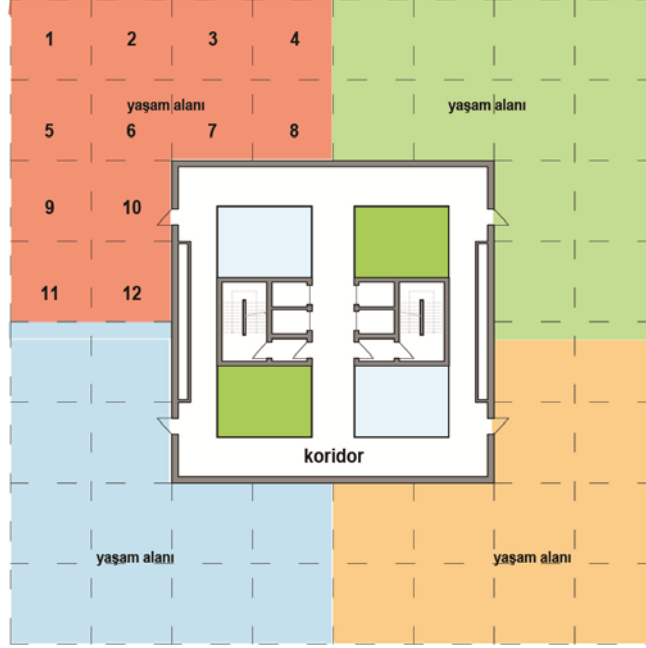
Belirtilen parametrelerin tamamlanması ile çekirdek alanının tasarlanması tamamlanmıştır. Çekirdek alanı tasarımının tamamlanmasından sonra modül birimlerinin oluşturulmasına geçilmiştir.

Temel modül

Hazırlanan modelde ana fikir tek bir modül üzerine kurulmuştur. Bütün türetmeler ile 2 ve 3 boyutlu tüm ürünler bir modülün evrimleşmesinden doğmaktadır. Bu nedenle temel modüle ait yaklaşımlar büyük bir öneme sahiptir. Oluşan bütün yaşam alanlarının tek bir modülden çoğalarak oluşması sayesinde hem rasyonel daire çözümlerine ulaşılmış hem de basit bir taşıyıcı ve yapı kurgusu sağlanmıştır. Bu sayede de model tarafından tasarlanan yapının tamamen rasyonel bir anlayışa sahip olması amaçlanmıştır.

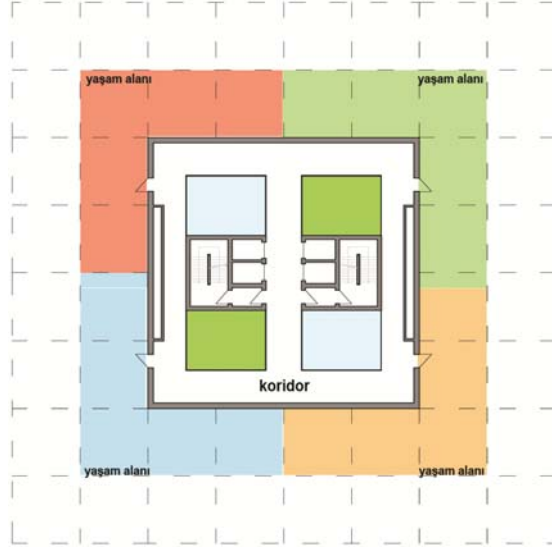
Projede çekirdek alanının etrafına 4 adet daire gelecek şekilde bir betik oluşturulmuştur. Öngörülen 4 daire için, maksimum büyüyecekleri bir sınır oluşturulmuştur. Bir daire en az 1 modül ile en fazla 12 modül arasındaki bir büyüklüğe sahip olabilmektedir. Bu kuralı belirlerken çekirdek alanının dört tarafına bir grid atanmış ve modüllerin bu grid içerisine yerleşmesi tanımlanmıştır. Bu sanal gride ait parametreler ile modüllere ait parametreler arasında doğru orantılı bir ilişki

kurulmuş ve böylece modüllerin veya gridin parametrelerindeki değişimlerin diğerinde de aynı anda gerçekleşmesi sağlanmıştır. Gridin arayüzde çekirdek alanına benzer bir şekilde tel örgü şeklinde görselleşebilmesi özelliği kullanıcıya bırakılmıştır. Dupleks daireler dışında tüm, daireler modüllerin aynı kat düzlemindeki bu gridi doldurması ile oluşmaktadır (Şekil 3.10).



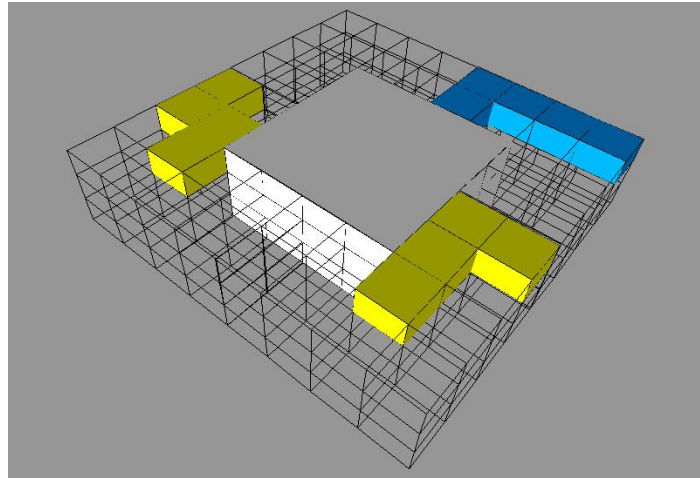
Şekil 3.10 : Modeldeki daire alanları.

Modüllere ait kurallar oluşturulurken belirlenen en önemli iki tanımlamadan ilki, model tarafından daire alanı içerisinde yaratılan ilk modülün çekirdek alanına komşuluk ilişkisiyle bağlı olması özelliğidir. Bu sayede dairelere ait ilk modüller daima çekirdek çevresinde oluşmaya başlamakta ve bu alanlar da dairelere giriş holleri olarak değerlendirilmektedir (Şekil 3.11).



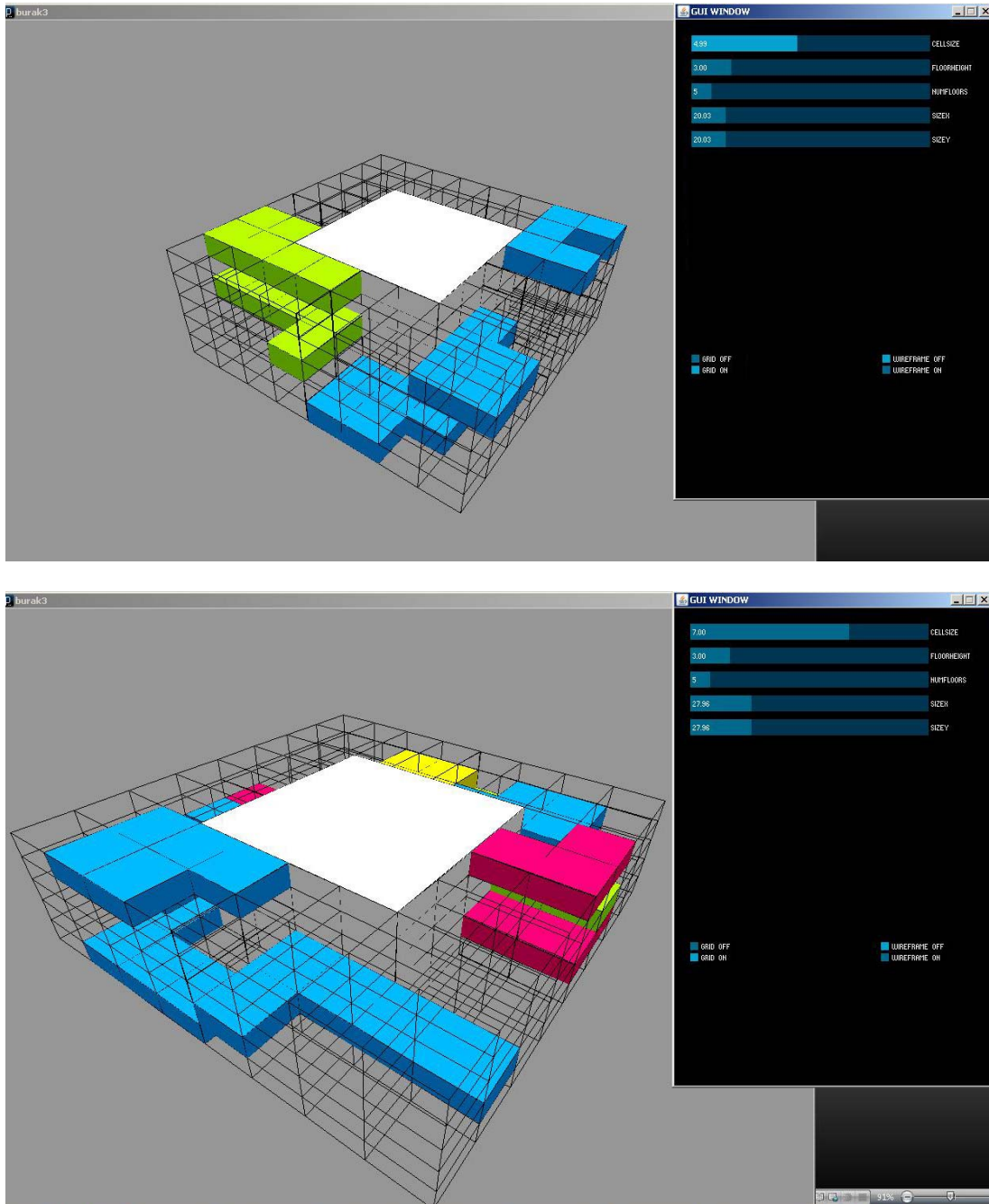
Şekil 3.11 : İlk modül yerleşim alanları.

İkinci tanımlama ise modüllerin çoğalması ile ilgilidir. Modüller çoğalırken daima bir önceki modülle kendi içinde komşuluk ilişkisine sahip olmakta ve bu sayede de daire planının sürekli olması ve hiçbir yerde kesintiye uğramaması sağlanmaktadır. Bu durum bütün potansiyel alanların tamamen doldurulduğu anlamına gelmemekte, yarı açık veya açık alanlara ait daireler oluşmakta fakat kapalı alanlar arası her zaman bir sıcak geçiş bağlantısı bulunmaktadır (Şekil 3.12).



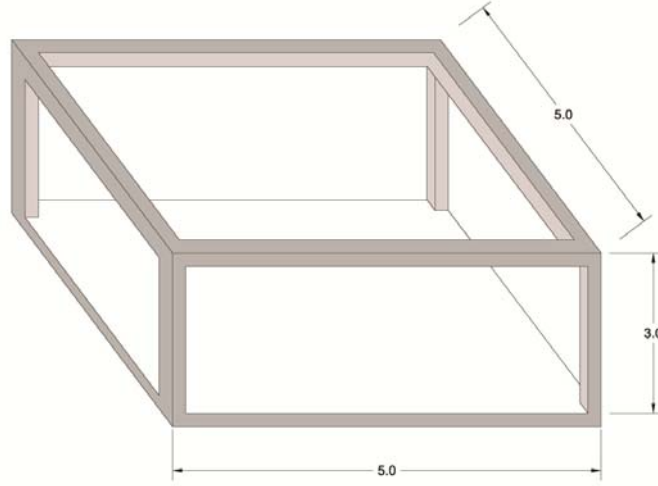
Şekil 3.12 : Modül yerleşimi ve kesintisizlik.

Temel modül ile ilgili kullanıcı kontrolüne bırakılan diğer parametre ise modül büyüklüğü ve yüksekliğidir. Modül yüksekliği kat yüksekliği ile aynı kontrol çubuğu haline getirilmiş böylece doğru orantılı bir şekilde büyümeleri sağlanmıştır. Modülü oluşturan diğer bir parametre olan büyüklük ise 0 metreden 10 metreye kadar olan bir aralıkta tanımlanmıştır. Kare planlı olarak tanımlanan temel modülün kenarları eşit şekilde büyümektedir (Şekil 3.13). Model kullanılırken önemli olan çekirdek büyüklüğü ile modül büyüklüğünün orantılı bir şekilde büyütülmesi gerekliliğidir. Bu sayede daha rasyonel planlı daire çözümleri elde edilmiş olunacaktır.



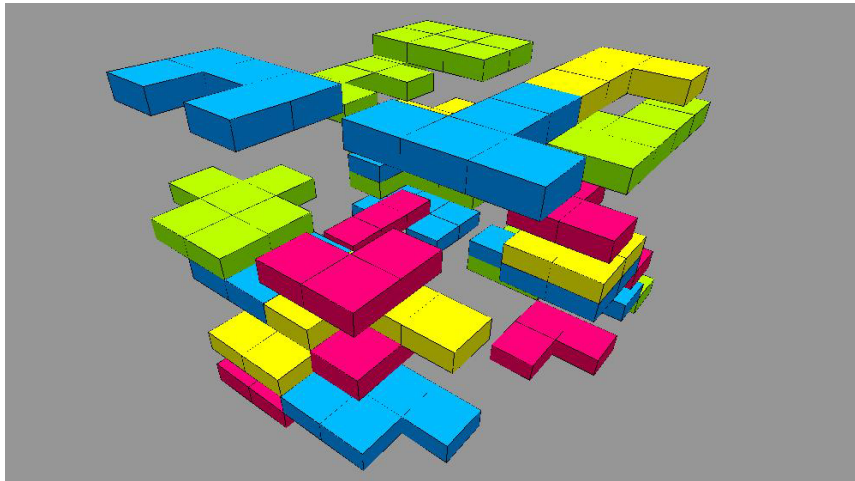
Şekil 3.13 : Modül büyüklük parametreleri.

Daha önce üzerinde durulduğu gibi model bünyesindeki optimal çekirdek boyutu 20 m. x 20 m. dir. Bu çekirdek boyutlarına uyan temel modül ebatları ise 5 m. x 5 m. x 3 m. olarak düşünülmektedir (Şekil 3.14). Betikten alınacak son ürünler bu ebattaki yapı birimleri üzerinden elde edilecektir.



Şekil 3.14 : Modül büyüklüğü.

Oluşturulan modelin en önemli taraflarından biri de ortaya çıkardığı ürünlerdeki varyasyon zenginliğidir. Dupleks daire olarak düşünülen iki katlı daire önerilerini hesaba almadığımızda dahi 12 adet gridin faktoriyel olarak olasılığı karşımıza plan şeması olarak gelebilmektedir. Bu sayede hem plan bazında hem de yapının bütününe düşündüğümüzde zengin bir perspektif algısı elde edilebilmektedir. Bu denli sayıda farklı plansal şemanın ortaya çıkması sayesinde her farklı tipte kullanıcı için özel daire planlarına ulaşılabilmektedir. Şekil 3.15’ de birkaç farklı modül birleşim şeması örneklenmektedir.



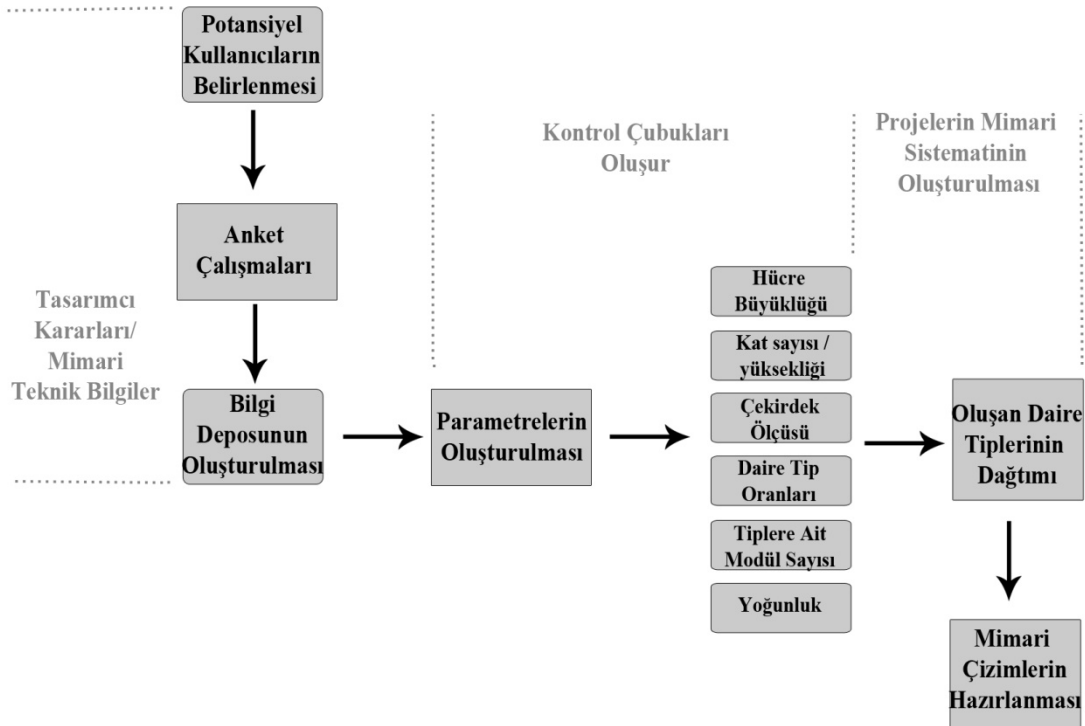
Şekil 3.15 : Modül birleşim şekilleri.

Temel modülü oluşturan parametrelerin tamamlanması ile daire tiplerinin oluşturulmasına geçilmiştir.

Daire tipolojileri

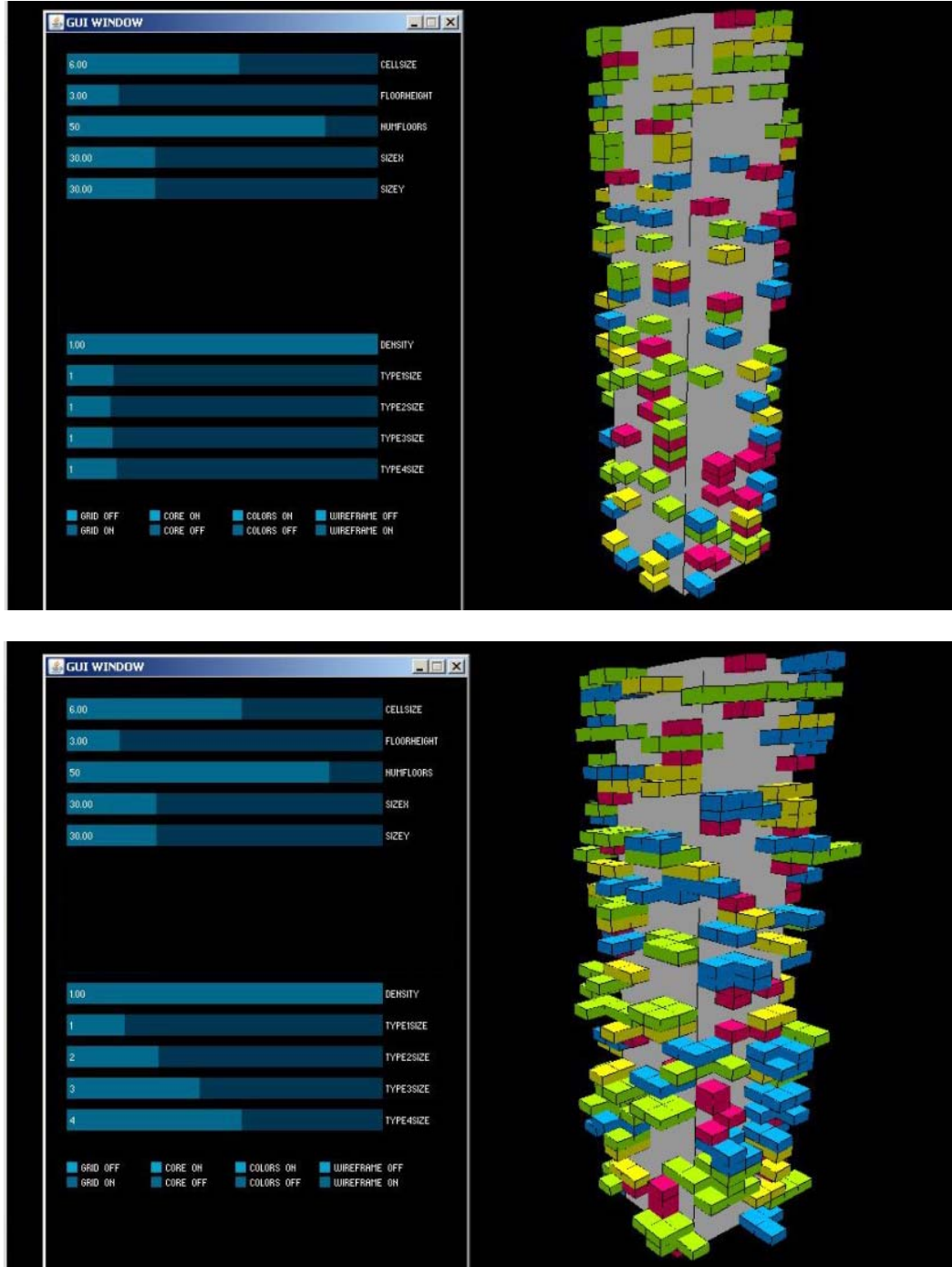
Bir önceki bölümde de belirttiğimiz gibi daireler farklı sayıda modülün birçok olasılık dâhilinde türemesi ile meydana gelmektedirler. Bu türetme işlemine ilk olarak daire tipleri için bir araya gelecek modül sayılarının belirlenmesi ile başlanır.

Daire tipolojileri oluşturulurken, proje öncesi muhtemel kullanıcılar arasında bir fizibilite çalışmasının yapıldığı varsayılmıştır. Bu fizibilite çalışmasında elde edilecek bilgiler ile hangi tip kullanıcıların projede yer alacağı, bu kullanıcıların sayıları, meslekleri, aile yapıları gibi bilgilerin elde edildiği öngörülmüştür. Bu bilgiler ışığında daire tiplerinin hangi oranda projede yer alması, daire tiplerinin ne kadar büyüklüklerde olması ve planlaması üzerinde durulduğu kabul edilmiştir. Kullanıcıların konut içerisinde bulunmasını istediği mekanların hacimsel büyüklükleri, galeriler, dubleks daireler gibi girdilerin bir bilgi havuzunda toplandığı öngörülmüştür. Bu bilgi havuzundan elde edilecek çıktılar ile de projenin gerçek kullanıcıları ve onların ihtiyaçlarının belirlenmesi amaçlanmıştır. Şekil 3.16'da yapılacak ön çalışmaları ve tasarım süreci arasındaki çalışma ağı görülebilmektedir.



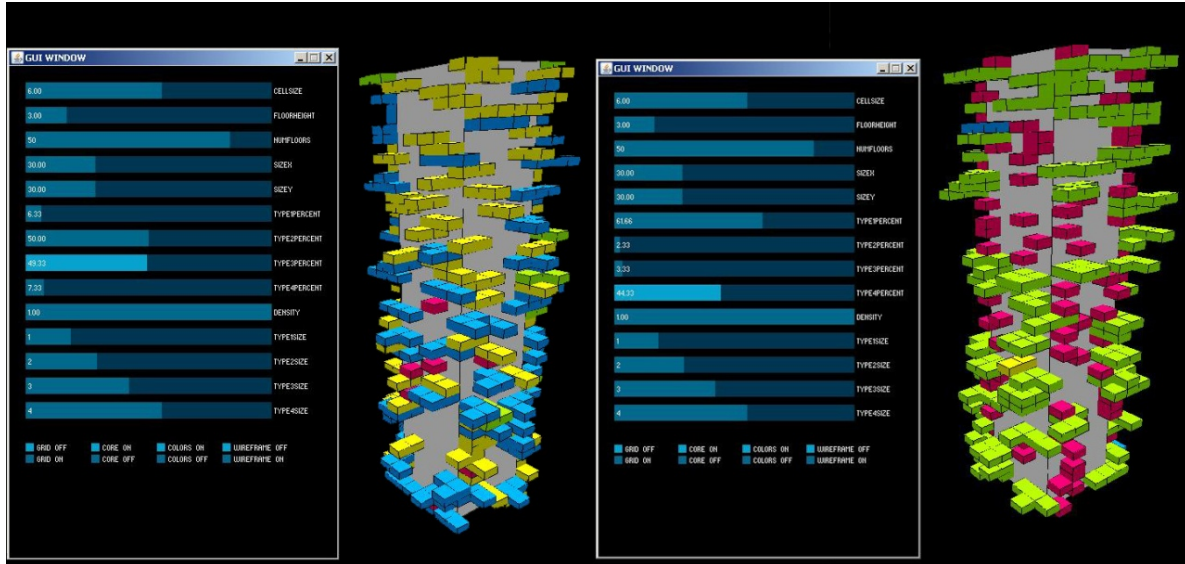
Şekil 3.16 : Fizibilite-Tasarım şeması.

Bu tanımlamalara göre toplamda 4 adet farklı tip daire büyüklüğü belirlenmiş ve hepsine farklı sayıda modül birliktelik ilişkisi atanmıştır. Bu birliktelikleri istersek geleneksel konut tasarımındaki tanımlamalar olan 1+1, 2+1, 3+1 ve 4+1 daireler olarak da kurgulayabiliriz. Bu dört farklı tip için sahip olabilecekleri modül sayıları parametresi betikte tanımlanmıştır. Bu sayede kullanıcı parametreleri değiştirerek her farklı tipte kaç adet modül olacağını kendi belirleyebilir. Bununla beraber arayüzdeki kontrol çubuklarını değiştirerek de bu ayarlamayı gerçekleştirebilir (Şekil 3.17).



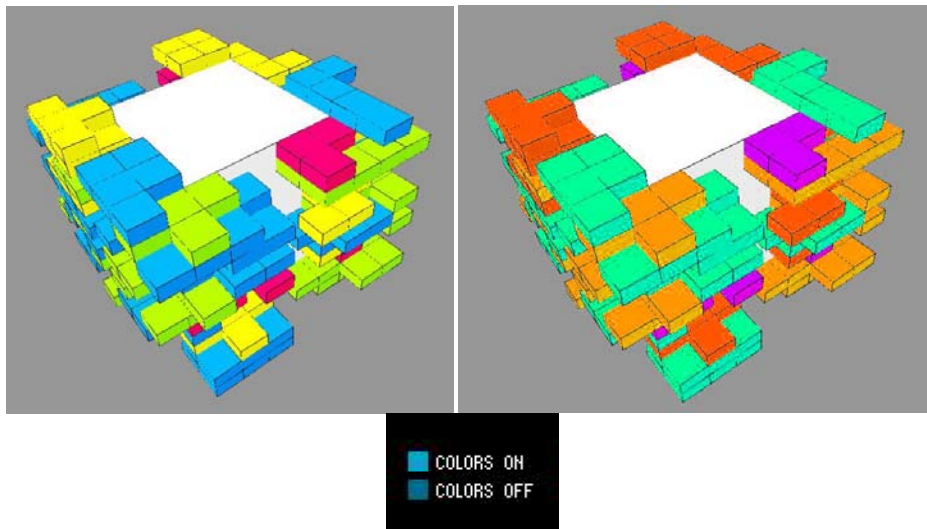
Şekil 3.17 : Tiplerdeki modül değişimleri.

Her bir tipin kaç modülden oluşacağını belirlemenmesi gibi bu tiplerin toplam daire sayısı içindeki oranları da betikte belirlenmiştir. Yüzdelik dilimde her tip için oranlar kullanıcılar tarafından değiştirilebilmektedir. Bu sayede kullanıcı hangi tipten yüzde kaçlık bir oranda dairenin projede yer alacağını belirleyebilmektedir (Şekil 3.18).



Şekil 3.18 : Tiplerin dağılım oranları.

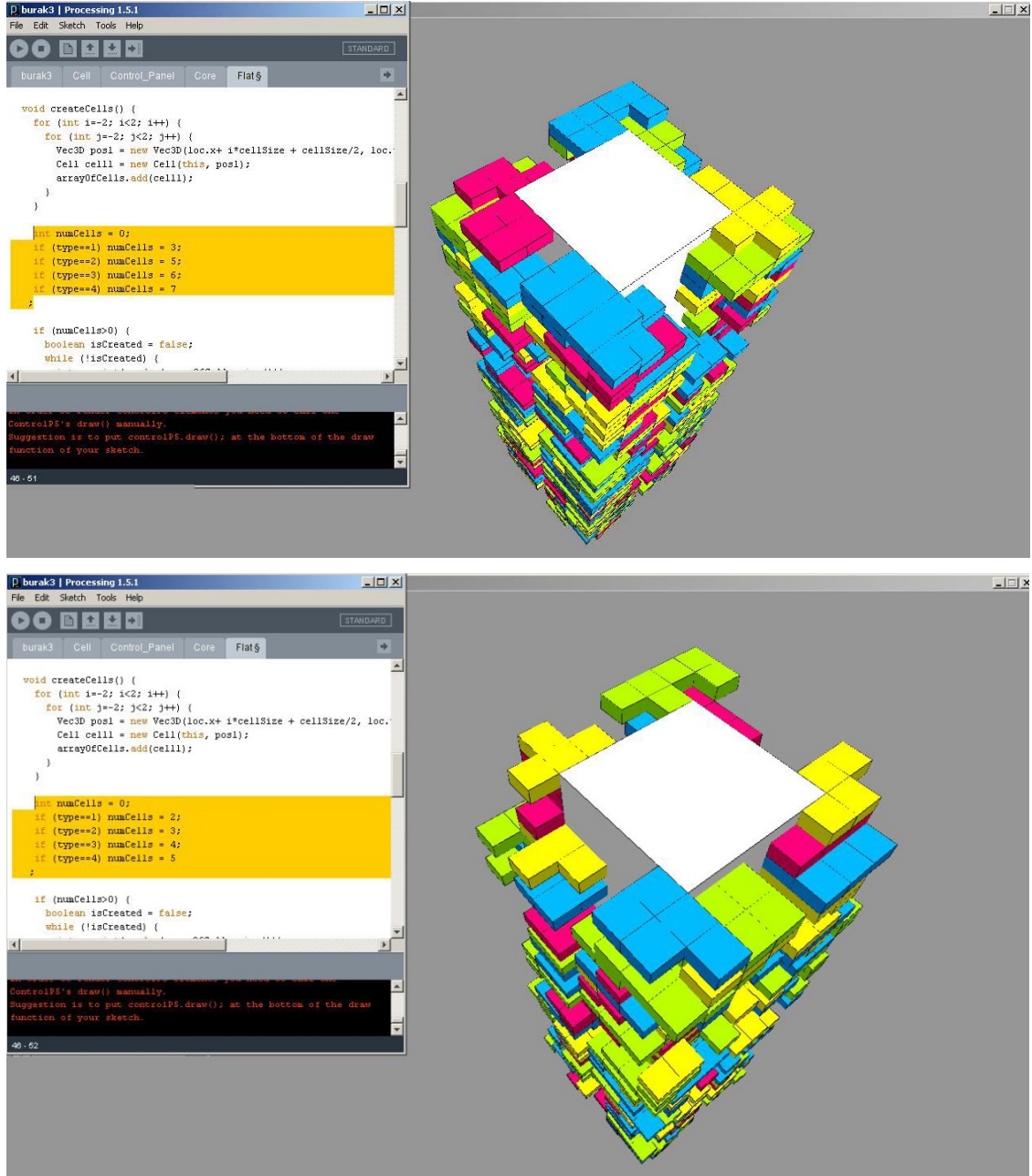
Daire tiplerinin rahatça seçilebilmesi için modelde her tip ayrı bir renkle belirtilmiştir. Bu renk seçimide kullanıcı tarafından değiştirilebilmektedir. Ayrıca renklerin istenildiği an kapatılabilmekte ve aynı renge dönülebilmektedir (Şekil 3.19).



Şekil 3.19 : Tiplerdeki renk değişimleri.

Algoritmada çekirdek çevresine gelecek olan bir modül her tip için bu sayılara eklenmektedir. Kısacası tip 1 çekirdek alanına komşu olan ilk modül ile artı 2 modül

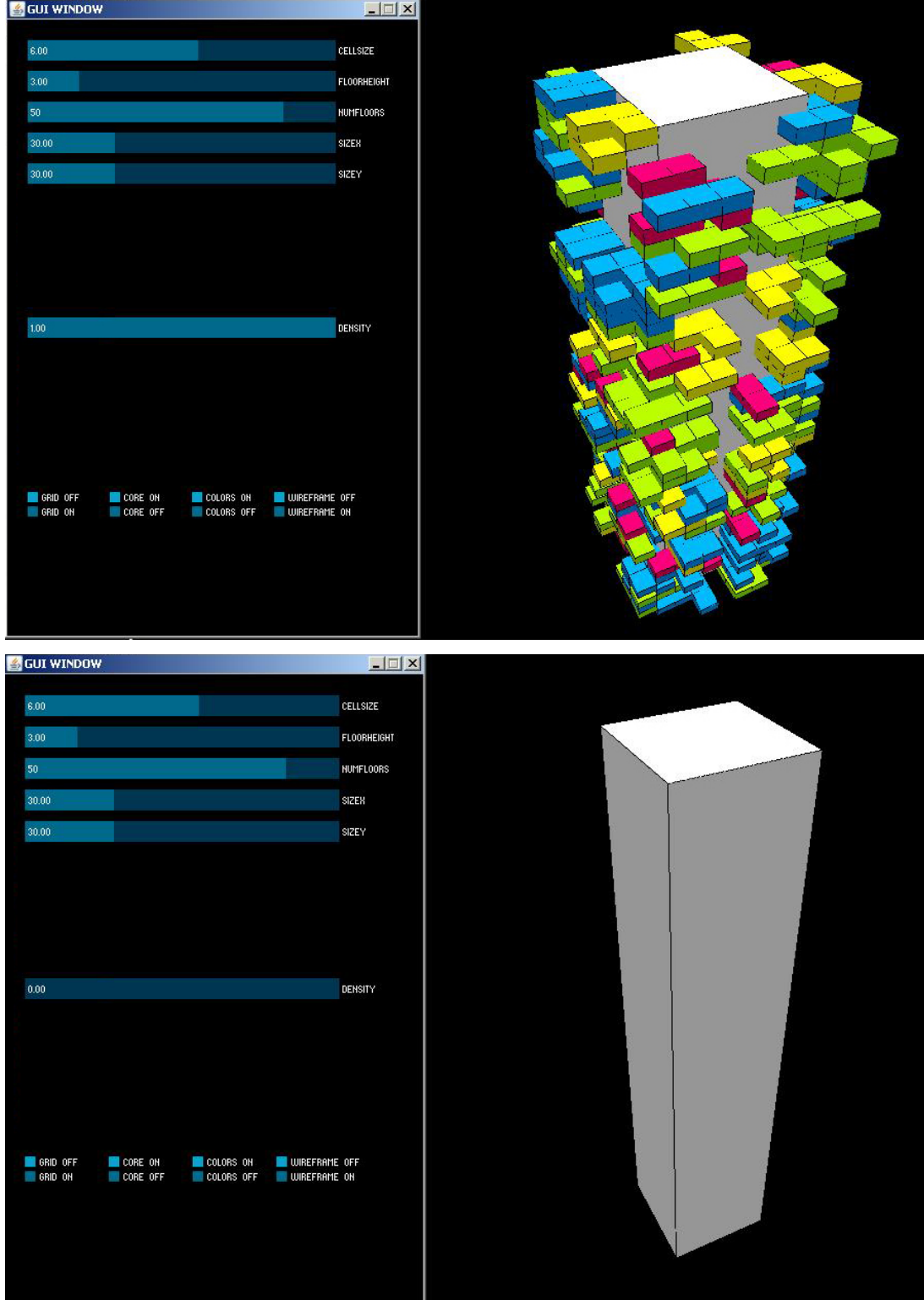
olmak üzere toplam 3 modülden oluşmaktadır. Her bir modülün 5 m. x 5 m. olarak 25 m² alana sahip olduğunu kabul edersek 1+1 daire 75 m², 2+1 daire 100 m², 3+1 daire 125 m², 4+1 daire ise 150 m² büyüklüktedir. 1+1 daireyi pembe, 2+1 daireyi sarı, 3+1 daireyi yeşil, 4+1 daireyi ise mavi renkte kabul edersek tiplere ait sayısal değişiklikleri Şekil 3.20’ de görebiliriz.



Şekil 3.20 : Tiplerdeki modül sayısı değişimleri.

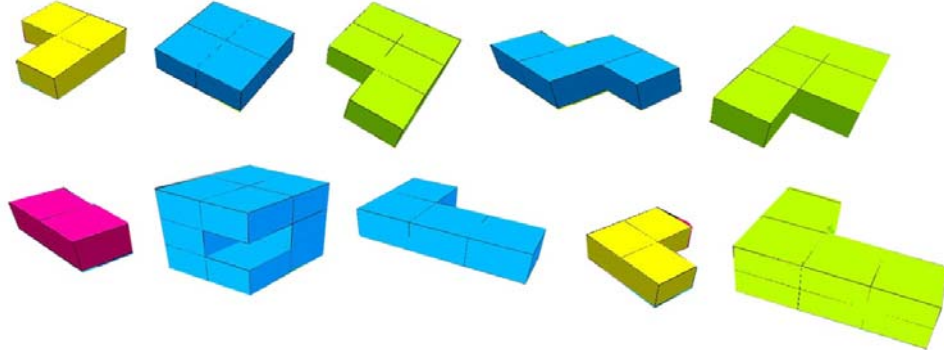
Daire tipolojilerinin oluşturulmasında kullanıcının kontrolüne bırakılan diğer bir parametre ise tiplerin projedeki yoğunluk dağılımlarıdır. Bu aşamada kullanıcı ilk olarak tüm projedeki yoğunluk katsayısını belirler. Bu katsayı 0 ile 1 arasında olup 0

iken tamamen boş, 1 iken ise bütün daire alanlarının doldurulduğu bir ürün ortaya çıkarır. Bu parametrenin kontrolü de arayüzdeki kontrol çubukları tarafından değiştirilebilmektedir. Toplam projedeki yoğunluk katsayısı belirlendikten sonra tiplerin dağılım oranlarının belirlenmesine geçilir (Şekil 3.21).



Şekil 3.21 : Daire yoğunlukları.

Daire tipolojileri ve tipolojilere ait modül sayıları oluşturulurken bazı yaşam senaryolarının varlığı önceden kabul edilmiştir. Bu yaklaşımda, tasarım süreci başlamadan önce birtakım araştırmaların yapıldığı ve bu araştırmalara göre de farklı ihtiyaçlara sahip konut kullanıcılarının olacağı öngörülmüştür. Örnekleme gerekirse günümüzdeki gibi daha az m²'ye sahip daireler az sayıda bireyi olan aileler için, da büyük daireler ise daha fazla bireyi olan aileler için düşünülmüştür. Modelin oluşturduğu birtakım alternatif daire tipleri Şekil 3.22 te görülebilmektedir.



Şekil 3.22 : Daire alternatifleri.

rnek olarak verilen bu daire tiplerine ait plan şemalarının kütleler ortaya çıktıktan sonra tasarımcı tarafından detaylandırılması gerekir. Bu işlem için plan şemaları mimari çizim programları kullanılarak oluşturulmalıdır. Tez kapsamında model tarafından oluşturulan birkaç daire şeması seçilerek, plan şemaları oluşturulmuştur. Bu tiplere ait plan şemaları Şekil 3.23' de görülebilir.



Şekil 3.23 : Daire plan çözümleri.

Planları tasarımcısı tarafından oluşturulan dairelerin genel olarak çekirdek alanı, şaftlar ve koridorlarla ilişkisini gösteren genel kat planı şeması Şekil 3.24’ de görülebilir.



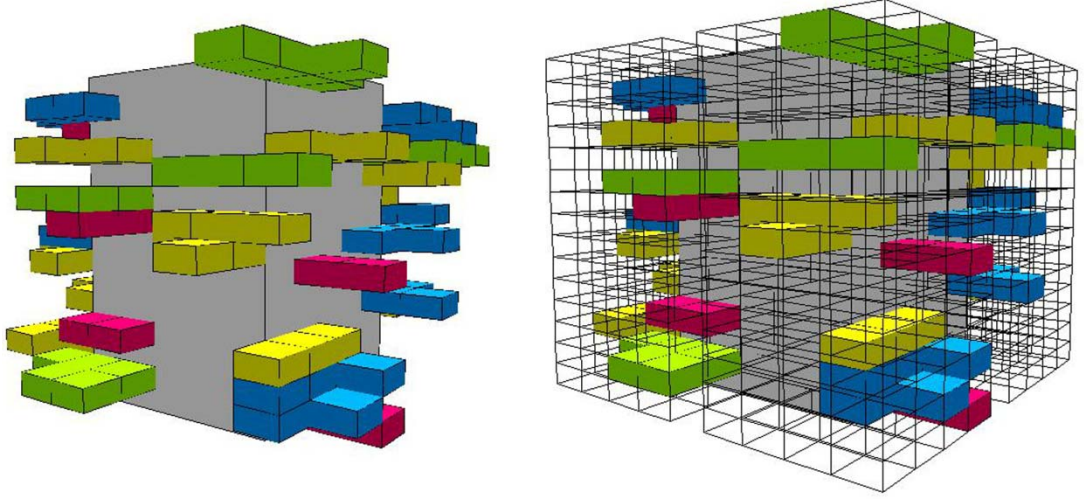
Şekil 3.24 : Genel kat planı.

Daire tiplerini oluşturan parametrelerin modele tanımlanması ile dış taşıyıcı akslarının ve modelin sahip olduğu diğer özelliklerin oluşturulmasına geçilmiştir.

Dış taşıyıcı aksları

Model ile oluşturulan konut projesinin taşıyıcı sisteminin asıl elemanı daha önceki bölümlerde de belirtildiği gibi çekirdeğin kendisidir. Çekidek dışındaki diğer taşıyıcı aksları ise modül büyüklüklerinin değişimine göre farklılık göstermektedir. Modelin ortaya koyduğu ürünlerde kütleli olarak taşıyıcı birimlerini görmesekte, 5 m.’lik kenar ebatlarına sahip modüler türetmenin dış aksları aralığı da 5’er metre olacak şekilde varsayılmıştır. Tüm türetmelerin tek bir modül üzerinden sağlanması sayesinde, modül boyutları ne kadar farklılaşırsa farklılaşsın, rasyonel ve orthogonal bir tartan grid oluşturulabilmektedir. Modüllere ait en kısa kenar boyutunun 5 metreden az olması durumunda ise iki modül uzunluğundaki mesafe taşıyıcı aksı olarak öngörülmüştür. Örnek vermek gerekirse, 8 m. x 8 m.

ebatlarındaki modüllerden oluşan bir sistemde dış akslar da 8 m. x 8 m. dir. Modül ebatlarının 3 m. x 3 m. olması durumunda ise iki modüle eş gelecek şekilde dış akslar 6 m. x 6 m. olarak düşünülür. Modül ebatlarının taşıyıcı aksı ile aynı değerde olduğu durumlarda, tel örgü özelliği kullanılarak taşıyıcı aksları da aynı anda görülebilir (Şekil 3.25).



Şekil 3.25: Taşıyıcı sistemleri.

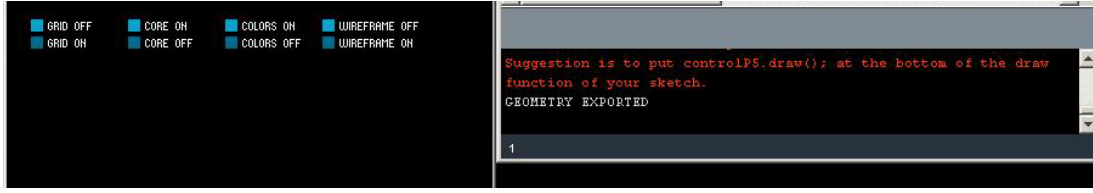
Şekil 3.25 da tüm tasarım alanına yayılmış taşıyıcı sistemleri görülmektedir. Modelden elde edilen projenin kolon-kiriş sistemi ile veya hücresel taşıma sistemi şeklinde ele alınması mümkündür. Bu yöntemle daire alanları konsollar oluşturmaktadır.

Model ve rhino programına transfer

Processing programı kullanılarak oluşturulan modelin, diğer programlar ile uyumlu bir şekilde çalışması ve farklı formatlara aktarılarak, farklı alternatifteki, daha ileri çalışmalarla desteklenebilmesi için export (dışa aktarım) özelliğine sahip olması gerekmektedir. Modelde tanımlanan algoritmanın sadece kütsel bazda daire şemaları ve yapı birimleri üretebilmesi, işlevsel ve estetik ihtiyaçları karşılaması gereken cephe karakterinin başka araçlar kullanılarak gerçekleştirilmesini gerekli kılmıştır. Modelin dışa aktarım özelliğine sahip olmasının diğer bir nedeni ise, üretilen tasarım ürününün görselleştirilmek istenmesidir.

Bu doğrultuda modelin Rhino programına atılması ve Rhino programından da diğer 3 boyutlu modelleme programlarına dönüştürülebilmesi sağlanmıştır. Modelin arayüzünde “e” tuşuna basıldığı zaman model tüm betiği text dosyası halinde Rhino

programına aktarmaktadır. Ayrıca kullanıcı “data” klasöründe bulunan text dosyasındaki betiği Rhino script sekmesine kopyalarak da bu işlemi gerçekleştirebilmektedir.

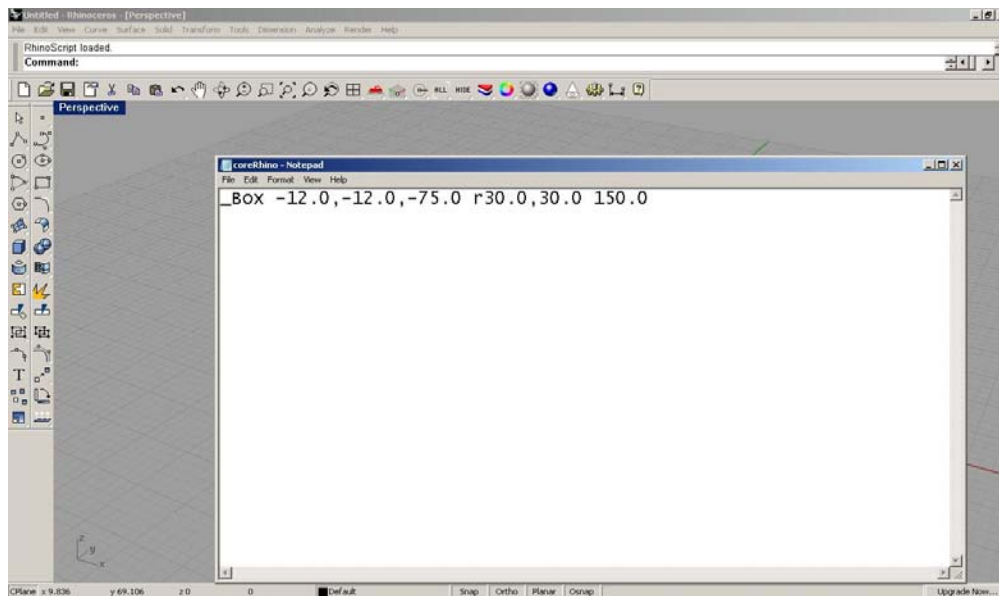


Şekil 3.26: Dışa aktarım.

Rhino programında açılan betik Rhino script sekmesinde okunmakta ve geometriyi oluşturmaktadır (Şekil 3.26). Oluşan geometri daha sonra 3 boyutlu modelleme programına atılarak, cephe elamanları tasarlanıp, görselleştirilebilmektedir.

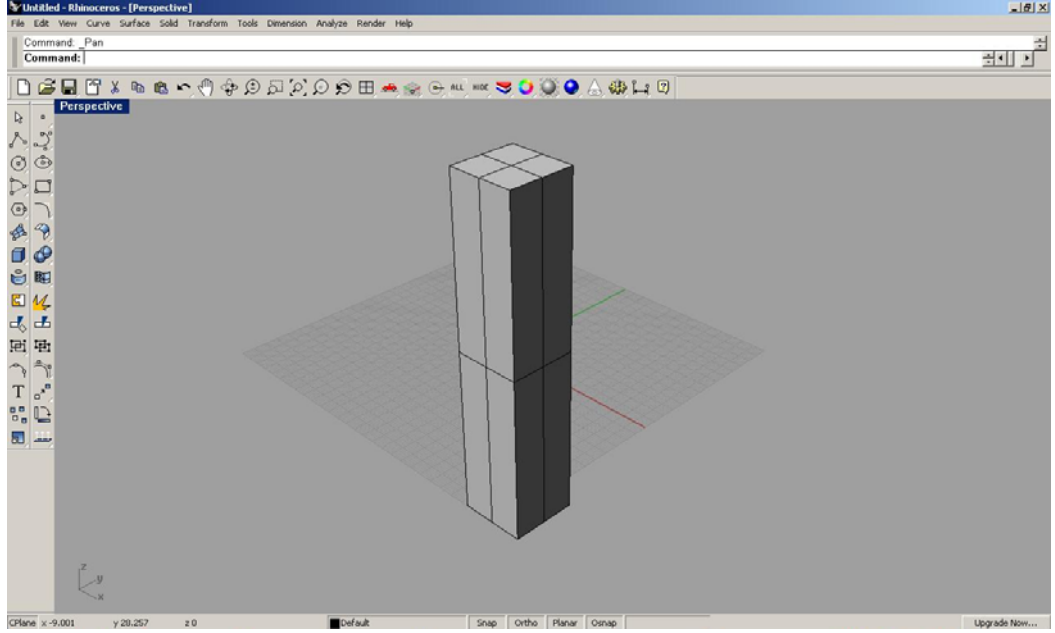
Geliştirilen model

Bu bölümde geliştirilen modelden daha önce belirlenmiş parametreler ile oluşturulan bir projenin, sırayla Rhino programına tanıtılması, cephe karakteristiğinin belirlenmesi ve daha sonra da görselleştirilmesinin anlatımı yapılacaktır. İlk olarak tasarlanan betik Rhino script formatına getirilerek bir “Notepad” dosyası formatında tekrar düzenlenmiştir. Buradaki amaç Rhino script formatının betiği sorunsuz bir şekilde tanımlayabilmesini sağlamaktır. Oluşturulan “Notepad” dosyasındaki betik kodu kopyalanarak Rhino programının “command” sekmesine yapıştırılır (Şekil 3.27).



Şekil 3.27 : Rhino-Çekirdek alanı betiği.

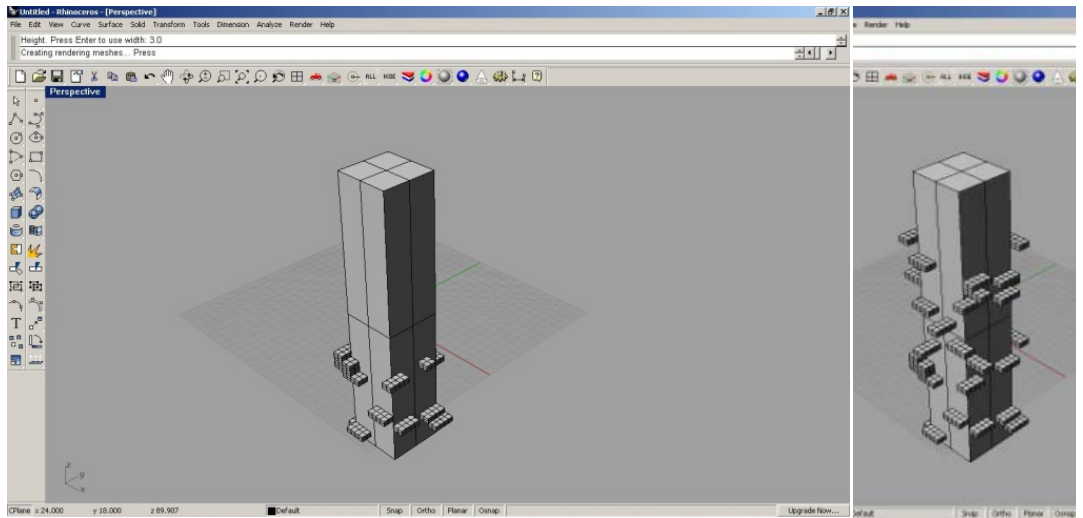
İlk olarak çekirdek alanını oluşturacak betik kopyalanmıştır. Bu kopyalama ile birlikte Rhino programında yüzeylerin oluştuğu görülebilir (Şekil 3.28).



Şekil 3.28: Rhino-Çekirdek alanı oluşumu.

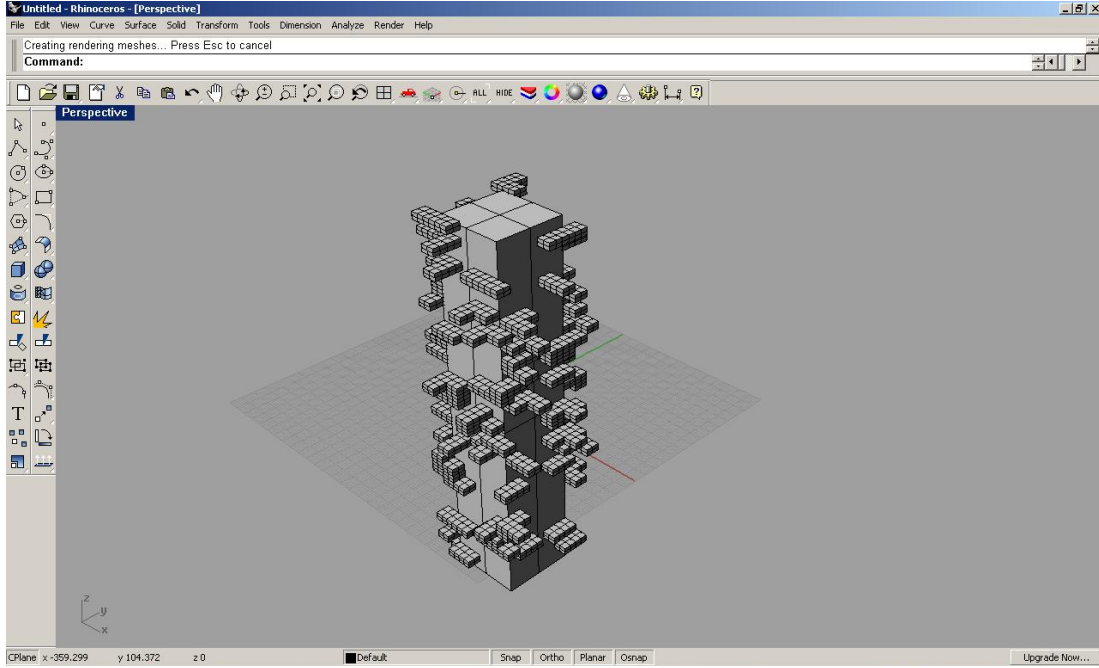
Çekirdek oluşumunda girilen parametreler, çekirdek ebatları için 30 m. x 30 m. kat yüksekliği 3 m. ve toplam kat sayısı 50 adettir. Notepad dosyasında da tanımlandığı gibi 0,0 ekseninde 150 m.lik bir kütle oluşturulmuştur.

Çekirdek yapısının oluşturulmasından sonra daire tiplerinin oluşturulmasına geçilmiştir. İlk olarak 1. dairelere ait betik “command” sekmesine girilmiştir (Şekil 3.29). Betikin girilmesi ile program bu tipe ait tüm daireleri oluşturmaya başlamıştır.



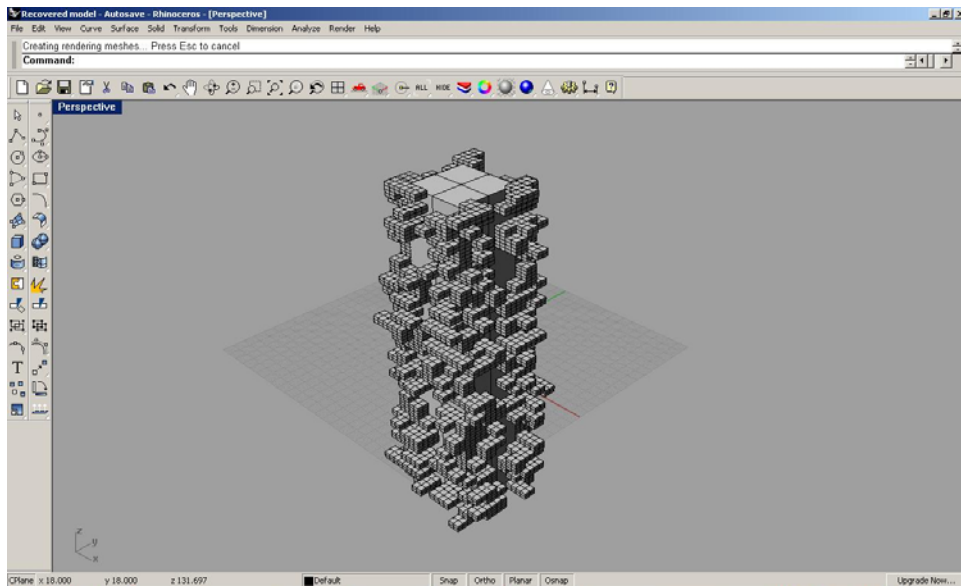
Şekil 3.29: Rhino-1.tip daireler.

1. tip dairelerin oluşturulmasından sonra aynı yöntemle 2. tip dairelerin oluşturulması ile devam edilmiştir. Command sekmesine 2. tiplere ait betikin girilmesi ile bu tip daireler oluşturulmuştur (Şekil 3.30).



Şekil 3.30: Rhino-2.tip daireler.

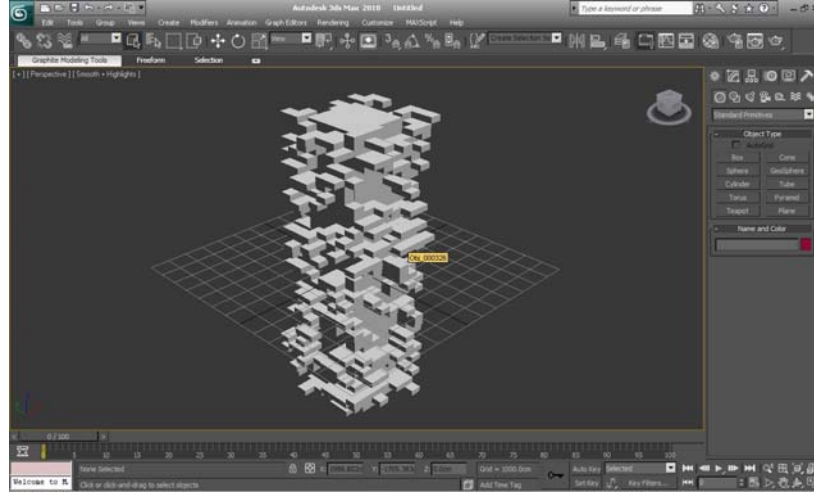
1. ve 2. Tip dairelerin oluşturulmasına benzer şekilde 3. ve 4. Tip dairelerde oluşturulmuş ve tüm dairelerin çekirdek etrafındaki konumlandırılmaları sonlanmıştır. Böylece oluşan kütlenin cephe kurgusunun tasarlanacağı 3ds max programına atılması için gereken süreç tamamlanmıştır (Şekil 3.31).



Şekil 3.31: Rhino-Tüm daireler.

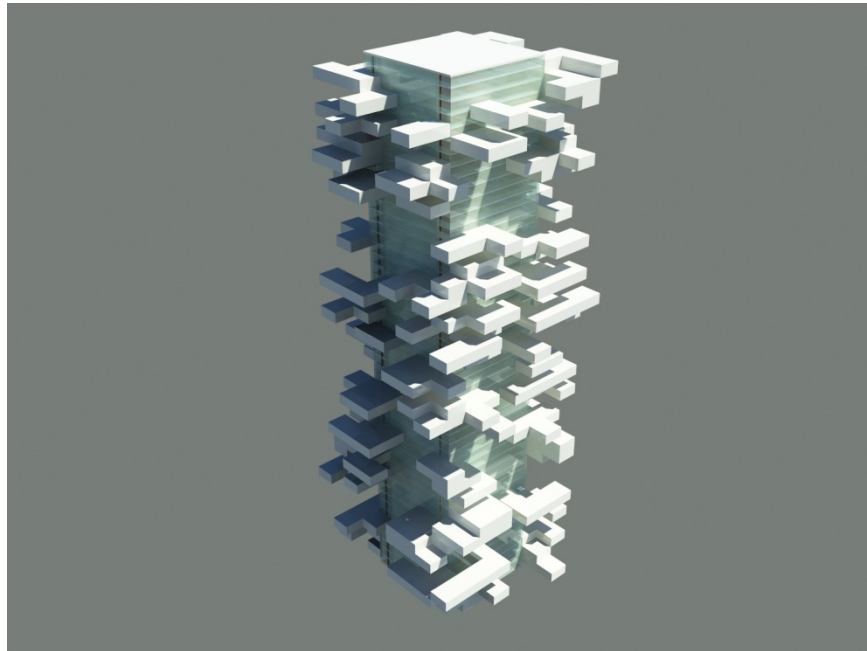
Daireler oluşturulurken ele alınan parametrelerden hücre uzunluğu ve genişliği 6 m. şeklindedir. Tüm daire tipleri %25 oranında eşit olarak yapıda dağıtılmıştır. Yoğunluk ise 0.7 katsayısı seçilerek belirlenmiş ve 1, 2, 3 ve 4. tip dairelere ait modül sayıları sırasıyla 3, 4, 5 ve 6 dır.

Oluşturulan model Rhino programında 3ds dosya formatı ile kaydedilerek, cephe tasarımı için 3ds max programına atılmıştır (Şekil 3.32).



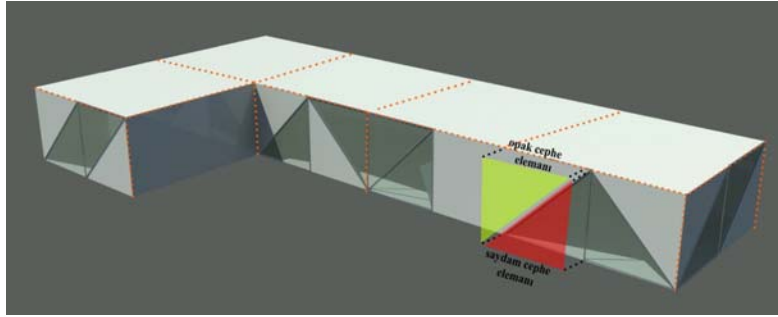
Şekil 3.32: Model - 3ds max.

Modelin tasarımının geliştirilmesi ve cephe kurgusunun oluşturulmasına ilk olarak çekirdek alanından başlanmıştır. Çekirdek alanı masif bir hacimden öteye kat döşemeleri ve sirkülasyon elemanları ile tekrar ele alınmıştır (Şekil 3.33).



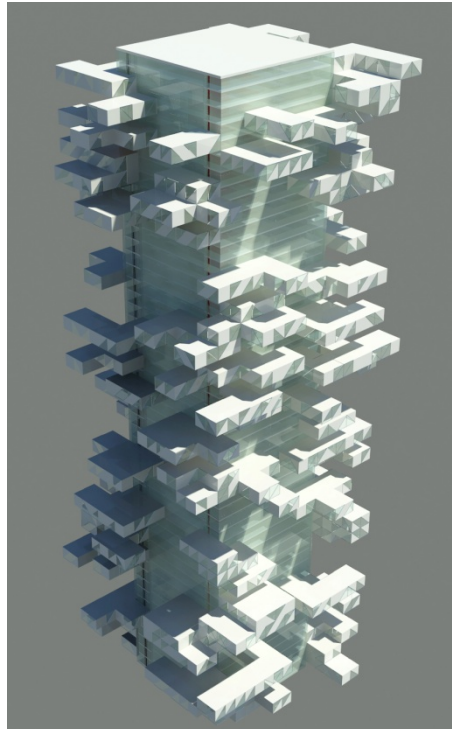
Şekil 3.33: Çekirdek-Modüller.

Çekirdek alanı mimarisi detaylı olarak ele alındıktan sonra daire tiplerinin cephe kurguları üzerinde çalışılmıştır. Cephe kurgusunun sistematığı kurgulanırken de hem alternatifli kat planları hem de kullanıcı istekleri doğrultusunda farklılaşabilen bir cephe kurgusu altlığı yaratılmak amaçlanmıştır. Bu bağlamda modüler kurgu üzerine çoğalan tasarımın, cephe tasarımında da modülerite kavramı üzerine yoğunlaşmıştır. Cephe kurgusu tasarlanırken bir modülün her yüzeyi iki köşe noktasının birleştirilmesi ile oluşan köşegen ile iki parçaya ayrılmıştır. Oluşan iki yüzeyin tasarımcının ve kullanıcının tercihi doğrultusunda saydam veya opak olacak şekilde kurgulanması amaçlanmıştır. Böylece her modül yüzeyinde farklılaşan bir cephe tasarımına gidilmiştir (Şekil 3.34).



Şekil 3.34: Cephe elemanları.

Cephe kurgusu her modülde aynı mantıkla ele alınmıştır. Tek modülden sonra tüm cephenin kurgulanması ile bütün tasarımdaki cepheyi ortaya çıkarmıştır (Şekil 3.35).



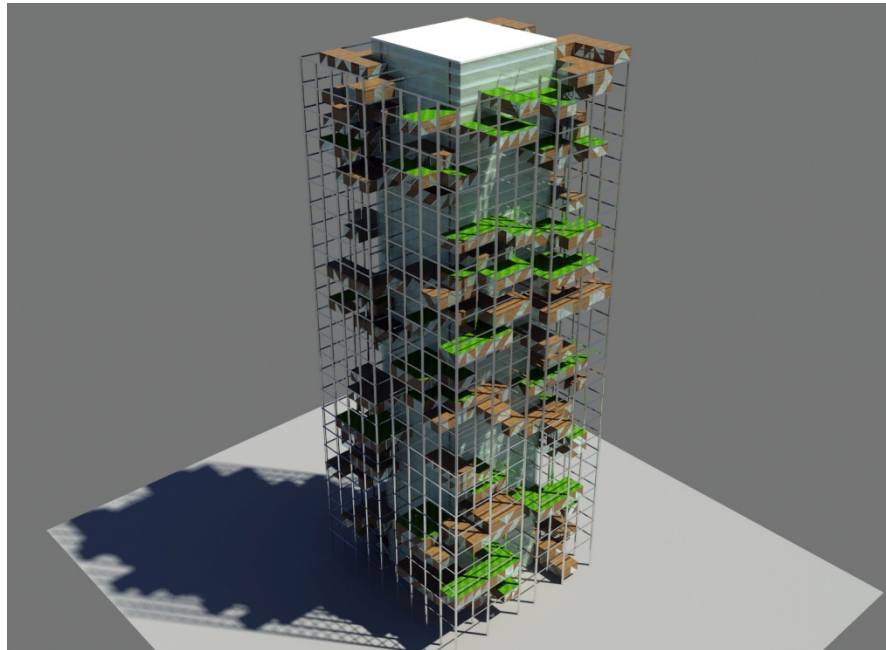
Şekil 3.35: Tüm cephe kurgusu.

Modüllere ait cephelerin oluşturulmasından sonra ise betiğin oluşturduğu taşıyıcı aksları dikkate alınarak taşıyıcı elemanlar yerleştirilmiştir. Yaşam birimleri arasında kalan açık ve yarı açık alanların dairelere hizmet eden teraslar olarak işlevlendirilmesi ile genel olarak yapının cephe tasarımı tamamlanmıştır (Şekil 3.36).



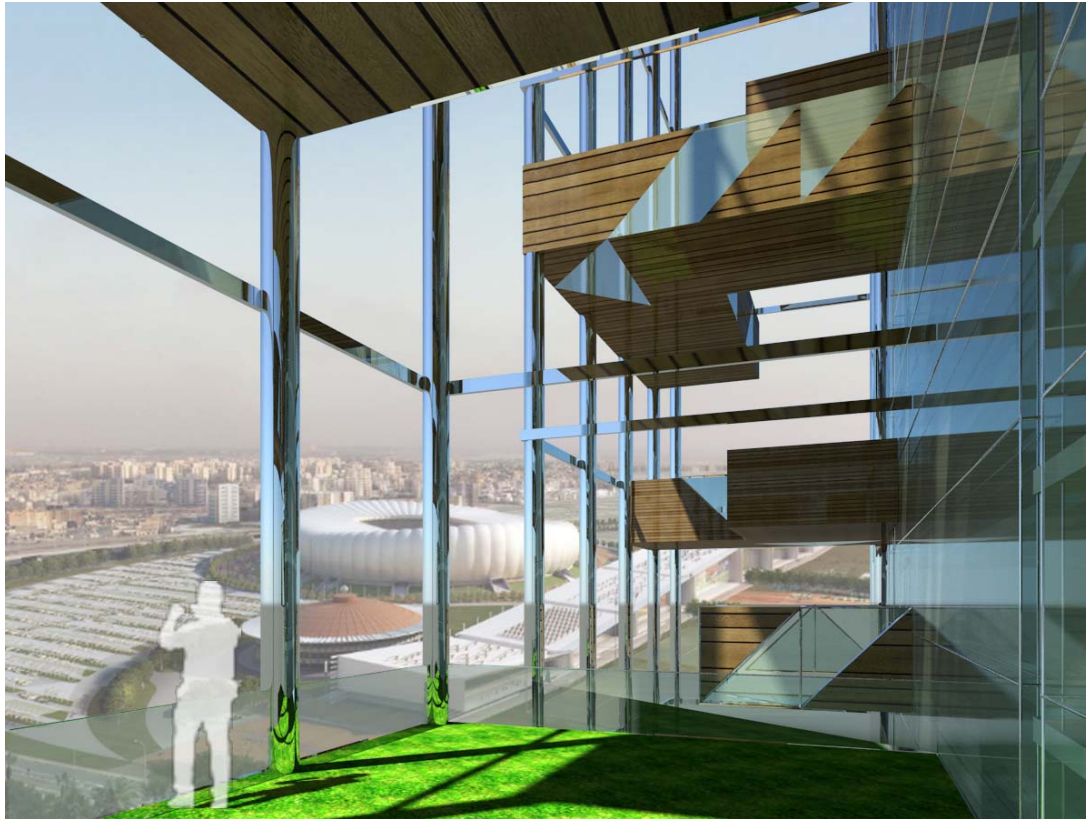
Şekil 3.36: Taşıyıcı ve teraslar.

Betik tarafından oluşturulan modelin görselleştirilmesi ile ilgili yapılan son işlev ise cephe malzemesinin modüllere atanmasıdır. Tüm tiplere cephe malzemeleri atanmış ve yapının tasarımı tamamlanmıştır (Şekil 3.37).



Şekil 3.37: Cephe malzemesi.

Betik tarafından oluşturulan farklı bakış açılarından modele ait görseller ise Şekil 3.38’de görülebilir.



Şekil 3.38: Modele ait kolajlar.

Bu grsellerin dıřında yapının hcresel tařıyıcı sistematięi ile ele alınma řekli de řekil 3.39’da grlebilir.



řekil 3.39: Hcresel tařıyıcı sistem.

4. SONUÇ VE ÖNERİLER

Bilişim teknolojilerinin her geçen gün daha hızlı ilerleyen gelişimini günümüzde mimarlık alanında daha etkili bir şekilde görmekteyiz. Daha karmaşık tasarım problemlerini çözebilmekte ve daha karışık ilişki ağlarını yorumlayabilmekteyiz. Bütün tasarım sürecinde, bilgisayar destekli çalışmalar ile daha önceleri cevap veremediğimiz çok bileşenli tasarım sorularına cevaplar verebilmekteyiz. Bilgisayar destekli tasarım süreçleri ile özellikle konut mimarisinde kullanıcı etkenlerini göz önüne alarak gelişen bir yaklaşım yöntemini daha etkili bir şekilde kullanabilme şansını elde etmiş durumdayız.

Tez kapsamında örnekler ve oluşturulan modellerle ele alınan, kullanıcısı özelinde gelişen modüler konut kavramı bu bağlamda her geçen gün daha önemli bir gereklilik haline gelmektedir. 20. Yüzyılın konut tasarım problemine yaklaşımını değiştirerek 21.yy teknolojisi ile yeni bir bakış açısı geliştirebilme şansını elimizde bulundurmaktayız. Bu anlamda konut tasarımında kullanıcı profili üzerine yoğunlaşan modeller üzerine araştırmalarımızı genişletmek durumundayız.

Tez kapsamında bu araştırmalara bir örnek olabilecek tasarım modelinin oluşturulması için örnekler üzerinden değerlendirmeler yapılmıştır. Bu değerlendirmeler üzerine yapılan kritikler ve değerlendirmelerden elde edilen bilgiler yeni modelin tasarlanmasında kullanılmıştır. Tasarlanan model ile toplu üretilen konut tasarım süreci farklı bir bakış açısı ile irdelenmiştir. Tasarımcının, istediği parametrelerle şekillendirebileceği ve her seferinde yüzlerce farklı tipte konut tasarımı üreten bir yazılım ile tasarım sürecinin zenginleştirilmesi ve geleneksel tasarım yaklaşımının kırılması amaçlanmıştır. Tüm süreç bilgisayar destekli betiklerin oluşturulması ve çalıştırılması ile yaratılsa da, mimari standartların ve rasyonel bilgi birikiminin çerçevesinde tasarım sonlandırılmıştır. Bu sayede konsept bir proje üretiminden ziyade, geliştirilebilir, rasyonel bir taşıyıcı sistemine sahip ve günümüz yapım teknolojisine kolayca adapte edilebilecek bir üretim şeması yaratılmıştır.

Çalışma dahilinde, başarılan hedefler ile birlikte tam anlamıyla gerçekleştirilemeyen hedefler de bulunmaktadır. Bu hedeflerden ilki fizibilite çalışmalarıdır. Tez çalışmasında daire tipolojileri oluşturulurken fizibilite çalışmalarının model dışında başka bir çalışma yöntemi olarak gerçekleştirildiği varsayılmıştır. Bu varsayım yerine modele atanacak bir bilgi toplayabilme ve değerlendirebilme özelliği ile bu fizibilite çalışmalarının da modele aktarılabilmesi ve modelin bu bilgilere göre daire tipi üretim seçim yapabilmesi, ilerdeki çalışmalarda modelde geliştirilecek noktalardan biridir. Bu sayede tasarımcının daireleri tek tek seçmesi yerine daha önce bilgilerin girildiği model kendi seçim işlemini gerçekleştirebilecektir.

Modelin geliştirilebilecek diğer bir özelliği ise kontrol çubuklarının ve buna bağlı parametrelerin arttırılması durumudur. Varolan parametreler ile beraber, modelde yapının tasarımında bulunacak kat bahçeleri ve balkonları atayabilecek yeni parametreler oluşturulabilir. Bunlar yanında dış taşıyıcı aksındaki düşey taşıyıcıların ebatlarının belirlenebileceği parametreler eklenebilir. Böylece artan kontrol çubuklarının sayısı ile tasarımcıya daha etkin bir kullanım ve karar hakkı verilebilir.

Modelde ileride üzerinde durulacak üçüncü nokta ise çekirdek alanının etrafındaki daire sayısının değişebilmesi durumudur. Tez kapsamındaki modelde çekirdek çevresinde dört adet daire bulunacak şekilde bir algoritma tanımlanmıştır. Bunun yerine çekirdek ebatlarının büyümesi veya küçülmesi ile doğru orantılı artan veya azalan kat daireleri oluşturulabilir. Bu sayede plan bazında kullanım dışı boş alanlarının önüne geçilmiş olunur.

Model tarafından tasarlanamayan ve başka programlar üzerinden kurgulanan diğer bir yapı elemanı cephedir. Tez kapsamında cephe kurgusu modelden alınan geometrinin 3 boyutlu modelleme programlarına atılması ile ayrı ortamlarda gerçekleştirilmiştir. İleri çalışmalarda cephe kurgusunun da tanımlanan algoritmalar ve yaratılan kontrol çubukları ile aynı yazılım bünyesinde gerçekleştirilmesi hedeflenmektedir. Örnek vermek gerekirse, kontrol çubukları yardımıyla cephede kullanılacak saydam ve opak panellerin sayıları ebatları ve iklim şartlarına göre yönlendirmeleri ayarlanabilir. Bununla birlikte renk ve saydamlık ayarları dahi bu yazılım içerisinde tanımlanabilir. Böylece model betiğinin başka yazılımlara atanması gerekliliğinin önüne geçilip, kullanıcısı tarafından basit kontrol çubukları ile tüm cephe karakteri tasarlanabilir.

Model algoritmasında eksik olan ve ilerideki çalışmalarda geliştirilmesi düşünülen son özellik ise çevre şartlarına göre değişimlerdir. Özellikle fiziksel çevre şartlarına göre değişebilen plan yerleşimleri, güneşin geliş açısı veya çevresel rüzgar etkilerine göre değişen plan kurgusu ve cephe karakteri üzerinde çalışılması hedeflenmektedir. Projenin gerçekleştirildiği bölgedeki iklim şartlarına bağlı saydamlık-opaklık oranı cephe kurgusu da diğer bir parametre olarak görülebilir.

Tez çalışması ve çalışma bünyesinde oluşturulan model sayesinde, yüzlerce alternatif daire şemasını rasyonel ve rahatça uygulanabilecek bir taşıyıcı sistemiği ile yaratılabileceği kanıtlanmış bulunmaktadır. Tüm katlarında aynı izdüşümde aynı daire tipleri ile gelişen konut bloklarının yerine farklı plan şemalarının bulunduğu, her kullanıcının kendisi için bir daire tipi seçebildiği bir tasarım modeli oluşturulabilmiştir. Genel kurgusu ile zengin perspektifler verebilen, ve geleneksel konut yaklaşımına alternatif bir duruş segileyen bir çalışma gerçekleştirilmiştir. Geleneksel metodlar yerine bilgisayar destekli sistemler ile tasarlanan bir proje sürecinin mümkün olabileceğinin gösterilmesi hedeflenirken, toplu üretilen konut tasarım süreçlerine alternatif bir metod eklemek amaçlanmıştır.

KAYNAKLAR

- Benros, D. ve Duarte, J. P.** (2008). An integrated system for providing mass customized housing. In *Automation in Construction 18* (Sf. 310–320). Lisbon, Portugal.
- El-Zanfaly.** (2009). Design by Algorithms: A generative design system for modular housing arrangement, Massachusetts Institute of Technology, USA.
- Kupferbeg, E.** (1974). Gropius Bauhausbauten Dessau, Laszlo Moholy-Nagy, Berlin, Germany.
- Kurokawa, Kisho.** (1993). New wave Japanese architecture, London, England.
- Torus, B.** (2008). Konut üretim sürecinde kural tabanlı bir yazılım modeli: Mardin örneği (19-22). Yıldız Technical University, İstanbul.
- Url-1** <http://www.west8.nl/projects/borneo_sporenbur>, alındığı tarih: 10.12.2011.
- Url-2** <http://housingprototypes.org/project?File_No=POR003>, alındığı tarih: 20.12.2011.
- Url-3** <<http://greatbuildings.com>>, alındığı tarih: 25.01.2012.
- Url-4** <<http://wikipedia.com>>, alındığı tarih: 29.02.2012.
- Url-5**<<http://inhabitat.com/sky-village>>,alındığı tarih: 12.03.2012.
- Url-6** <<http://inhabitat.com/t-tree-a-towerin>>, alındığı tarih: 23.03.2012.
- Url-7** <[http:// www.barcodehousing.net](http://www.barcodehousing.net)>, alındığı tarih: 18.03.2012.

EKLER

EK A : Betikler

EK A

Betik A.1: Tasarım Alanı.

```
// SETUP -----  
  
void setup() {  
  
  size(1024, 768, P3D);  
  
  cam = new PeasyCam(this, 300);  
  
  controlPanel();  
  
  core1 = new Core(new Vec3D(0, 0, 0));}  
  
// DRAW -----  
  
void draw() {  
  
  background(150);  
  
  ambientLight(150, 150, 150);  
  
  directionalLight(255, 255, 255, -1, 0.5, -1);  
  
  core1.run();}
```

Betik A.2: Çekirdek.

```
class Core {  
  
  // CLASS PARAMETERS -----  
  
  Vec3D loc;  
  
  ArrayList arrayOfFlats = new ArrayList();  
  
  // CONSTRUCTOR -----  
  
  Core(Vec3D _loc) {  
  
    loc = _loc;  
  
    createFlats();  
  
  }  
  
  // FUNCTIONS -----  
  
  void run() {  
  
    pushMatrix();  
  
    translate(0, 0, numFloors*floorHeight/2);  
  
    display();  
  
  }  
  
}
```

```

displayFloors();

runFlats();

popMatrix(); } //-----

void createFlats() {

    for (int i=0; i<numFloors; i++) {

        Vec3D    pos1    =    new    Vec3D(loc.x-sizeX/2,    loc.y-sizeY/2,    i*floorHeight-
(numFloors*floorHeight/2));

        Flat flat1 = new Flat(this, pos1);

        arrayOfFlats.add(flat1);

Vec3D pos2 = new Vec3D(loc.x+sizeX/2, loc.y-sizeY/2, i*floorHeight-(numFloors*floorHeight/2));

        Flat flat2 = new Flat(this, pos2);

        arrayOfFlats.add(flat2);

Vec3D pos3 = new Vec3D(loc.x-sizeX/2, loc.y+sizeY/2, i*floorHeight-(numFloors*floorHeight/2));

        Flat flat3 = new Flat(this, pos3);

        arrayOfFlats.add(flat3);

Vec3D pos4 = new Vec3D(loc.x+sizeX/2, loc.y+sizeY/2, i*floorHeight-(numFloors*floorHeight/2));

        Flat flat4 = new Flat(this, pos4);

        arrayOfFlats.add(flat4);

    }

}

//-----

void runFlats() {

    for (int i=0; i<arrayOfFlats.size(); i++) {

        Flat f = (Flat) arrayOfFlats.get(i);

        f.run();

    }

}

//-----

void displayFloors() {

    if (displayGrid) {

```

```

    for (int i=0; i<numFloors; i++) {

        pushMatrix();

        translate(loc.x, loc.y, i*floorHeight - floorHeight*numFloors/2);

        noFill();

        stroke(0);

        strokeWeight(1);

        box(sizeX, sizeY, 0);

        popMatrix();

    }

}

//-----

void display() {

    if (displayCore) {

        pushMatrix();

        translate(loc.x, loc.y, loc.z);

        fill(255);

        if (isWireframe) noFill();

        stroke(0);

        strokeWeight(1);

        box(sizeX, sizeY, numFloors*floorHeight);

        popMatrix();

    }

}

//-----

}

```

Betik A.3: Temel Modül.

```

class Cell {

// CLASS PARAMETERS -----

```

```

Flat parent;

Vec3D loc;

boolean exists = false;

boolean isCore = false;

// CONSTRUCTOR -----

Cell(Flat _parent, Vec3D _loc) {

    parent = _parent;

    loc = _loc;


    if (loc.x<parent.parent.loc.x+sizeX/2) {

        if (loc.x>parent.parent.loc.x-sizeX/2) {

            if (loc.y<parent.parent.loc.y+sizeY/2) {

                if (loc.y>parent.parent.loc.y-sizeY/2) {

                    isCore = true;

                }

            }

        }

    }

}

// FUNCTIONS -----

void run() {

    displayGrid();

    display();

}

//-----

void displayGrid() {

    if (displayGrid && !isCore) {

        pushMatrix();

        translate(loc.x, loc.y, loc.z);

        noFill();

```

```

    stroke(0);

    strokeWeight(1);

    box(cellSize, cellSize, floorHeight);
popMatrix();

}

}

//-----

void display() {

    if (exists) {

        pushMatrix();

        translate(loc.x, loc.y, loc.z);

        stroke(0);

        strokeWeight(1);


        if (parent.type==1) fill(255, 0, 100);
        if (parent.type==2) fill(255, 255, 0);
        if (parent.type==3) fill(0, 150, 255);
        if (parent.type==4) fill(150, 255, 0);
        if(!displayColors) fill(255);
        if (isWireframe) noFill();


        box(cellSize, cellSize, floorHeight);

        popMatrix();

    }

}

```

Betik A.4: Daire Renk.

```

if (parent.type==1) fill(255, 0, 100);
if (parent.type==2) fill(255, 255, 0);
if (parent.type==3) fill(0, 150, 255);

```

```

if (parent.type==4) fill(150, 255, 0);

if(!displayColors) fill(255);

if (isWireframe) noFill();

```

Betik A.5: Modül Sayısı.

```

int numCells = 0;

if (type==1) numCells = 2;

if (type==2) numCells = 3;

if (type==3) numCells = 4;

if (type==4) numCells = 5

```

Betik A.6: Daire Tipleri.

```

class Flat {

// CLASS PARAMETERS -----

Core parent;

Vec3D loc;

int type = 0; // 0-green, 1-(1+1), 2-(2+1), 3-(3+1), 4-(4+1)

ArrayList arrayOfCells = new ArrayList();

// CONSTRUCTOR -----

Flat(Core _parent, Vec3D _loc) {

parent = _parent;

loc = _loc;

float r = random(type1percent + type2percent + type3percent + type4percent);

if (r<type1percent) type = 1;

if (r>type1percent && r<type1percent+type2percent) type = 2;

if (r>type1percent+type2percent && r<type1percent+type2percent+type3percent) type = 3;

if

(r>type1percent+type2percent+type3percent
r<type1percent+type2percent+type3percent+type4percent) type = 4;

float d = random(1);

if (d>density) type = 0;

```

```

createCells();

}

// FUNCTIONS -----

void run() {
runCells();
}

//-----

void createCells() {
for (int i=-2; i<2; i++) {
for (int j=-2; j<2; j++) {

Vec3D pos1 = new Vec3D(loc.x+ i*cellSize + cellSize/2, loc.y + j*cellSize + cellSize/2,
loc.z+floorHeight/2);

Cell cell1 = new Cell(this, pos1);
arrayOfCells.add(cell1);
}
}

int numCells = 0;

if (type==1) numCells = type1size;
if (type==2) numCells = type2size;
if (type==3) numCells = type3size;
if (type==4) numCells = type4size;

if (numCells>0) {
boolean isCreated = false;
while (!isCreated) {
int r = int(random(arrayOfCells.size()));
Cell c = (Cell) arrayOfCells.get(r);
if (!c.isCore) {
for (int j=0; j<arrayOfCells.size(); j++) {
Cell other = (Cell) arrayOfCells.get(j);

```



```

if (other.isCore) {

float distance = c.loc.distanceTo(other.loc);

if ( distance<cellSize+0.1) {

c.exists = true;

isCreated = true;

break;

}

}

}

}

}

}

}

}

for (int i=0; i<numCells-1; i++) {

while (true) {

int r = int(random(arrayOfCells.size()));

Cell c = (Cell) arrayOfCells.get(r);

if (!c.isCore && !c.exists) {

float minDist = 1000000;

for (int j=0; j<arrayOfCells.size(); j++) {

Cell other = (Cell) arrayOfCells.get(j);

if (other.exists) {

float distance = c.loc.distanceTo(other.loc);

if ( distance<minDist) {

minDist = distance;

}

}

}

}

if (minDist<cellSize+0.1) {

c.exists = true;

break;

```

```

}

}

}

}

}

//-----

void runCells() {

for (int i=0; i<arrayOfCells.size(); i++) {

Cell c = (Cell) arrayOfCells.get(i);

c.run();

}

}

//-----

}

```

Betik A.7: Dış Taşıyıcı Aksı.

```

boolean displayGrid = true;

boolean displayCore = true;

boolean displayColors = true;

boolean isWireframe = false;

```

Betik A.8: Export.

```

void export () {

output = createWriter("data/allRhino.txt");

for (int j=0; j<core1.arrayOfFlats.size(); j++) {

Flat f = (Flat) core1.arrayOfFlats.get(j);

for (int i=0; i<f.arrayOfCells.size(); i++) {

Cell c = (Cell) f.arrayOfCells.get(i);

if (c.exists && !c.isCore) {

output.println("_Box " + c.loc.x + "," + c.loc.y + "," + c.loc.z + " r" + cellSize + "," + cellSize + " " +
floorHeight);

}

}
}
}

```

```

}

}

output.flush();

output.close();

output = createWriter("data/type1Rhino.txt");

for (int j=0; j<core1.arrayOfFlats.size(); j++) {

Flat f = (Flat) core1.arrayOfFlats.get(j);

for (int i=0; i<f.arrayOfCells.size(); i++) {

Cell c = (Cell) f.arrayOfCells.get(i);

if (c.exists && !c.isCore && c.parent.type==1) {

output.println("_Box " + c.loc.x + "," + c.loc.y + "," + c.loc.z + " r" + cellSize + "," + cellSize + " " +
floorHeight);

}

}

}

output.flush();

output.close();

output = createWriter("data/type2Rhino.txt");

for (int j=0; j<core1.arrayOfFlats.size(); j++) {

Flat f = (Flat) core1.arrayOfFlats.get(j);

for (int i=0; i<f.arrayOfCells.size(); i++) {

Cell c = (Cell) f.arrayOfCells.get(i);

if (c.exists && !c.isCore && c.parent.type==2) {

output.println("_Box " + c.loc.x + "," + c.loc.y + "," + c.loc.z + " r" + cellSize + "," + cellSize + " " +
floorHeight);

}

}

}

output.flush();

output.close();

output = createWriter("data/type3Rhino.txt");

```

```

for (int j=0; j<core1.arrayOfFlats.size(); j++) {
Flat f = (Flat) core1.arrayOfFlats.get(j);
for (int i=0; i<f.arrayOfCells.size(); i++) {
Cell c = (Cell) f.arrayOfCells.get(i);
if (c.exists && !c.isCore && c.parent.type==3) {
output.println("_Box " + c.loc.x + "," + c.loc.y + "," + c.loc.z + " r" + cellSize + "," + cellSize + " " +
floorHeight);
}
}
}
output.flush();
output.close();
output = createWriter("data/type4Rhino.txt");
for (int j=0; j<core1.arrayOfFlats.size(); j++) {
Flat f = (Flat) core1.arrayOfFlats.get(j);
for (int i=0; i<f.arrayOfCells.size(); i++) {
Cell c = (Cell) f.arrayOfCells.get(i);
if (c.exists && !c.isCore && c.parent.type==4) {
output.println("_Box " + c.loc.x + "," + c.loc.y + "," + c.loc.z + " r" + cellSize + "," + cellSize + " "
+ floorHeight);
}
}
}
output.flush();
output.close();
output = createWriter("data/coreRhino.txt");
output.println("_Box " + ((-sizeX/2)+(cellSize/2)) + "," + ((-sizeY/2)+(cellSize/2)) + "," + -
numFloors*floorHeight/2 + " r" + sizeX + "," + sizeY + " " + numFloors*floorHeight);
output.flush();
output.close();
println("GEOMETRY EXPORTED");

```

```

}

//-----

void keyPressed() {

if (key == 'e' || key == 'E') {

export();

}

}

```

Betik A.9: Kontrol Çubukları.

```

void controlPanel(){

control = new ControlP5(this);

GUIwindow = control.addControlWindow("GUI WINDOW",100,100,400,600);

GUIwindow.hideCoordinates();

myListener = new MyControlListener();

Slider s1 = control.addSlider("cellSize",1,10,cellSize,20,20,300,20);

s1.setWindow(GUIwindow);

s1.addListener(myListener);

Slider s2 = control.addSlider("floorHeight",2,8,floorHeight,20,50,300,20);

s2.setWindow(GUIwindow);

s2.addListener(myListener);

Slider s3 = control.addSlider("numFloors",1,60,numFloors,20,80,300,20);

s3.setWindow(GUIwindow);

s3.addListener(myListener);

Slider s4 = control.addSlider("sizeX",10,80,sizeX,20,110,300,20);

s4.setWindow(GUIwindow);

s4.addListener(myListener);

Slider s5 = control.addSlider("sizeY",10,80,sizeY,20,140,300,20);

s5.setWindow(GUIwindow);

s5.addListener(myListener);

Slider s6 = control.addSlider("type1percent",0,100,type1percent,20,170,300,20);

s6.setWindow(GUIwindow);

```

```

s6.addListener(myListener);

Slider s7 = control.addSlider("type2percent",0,100,type2percent,20,200,300,20);
s7.setWindow(GUIwindow);
s7.addListener(myListener);

Slider s8 = control.addSlider("type3percent",0,100,type3percent,20,230,300,20);
s8.setWindow(GUIwindow);
s8.addListener(myListener);

Slider s9 = control.addSlider("type4percent",0,100,type4percent,20,260,300,20);
s9.setWindow(GUIwindow);
s9.addListener(myListener);

Slider s10 = control.addSlider("density",0,1,density,20,290,300,20);
s10.setWindow(GUIwindow);
s10.addListener(myListener);

Slider s11 = control.addSlider("type1size",0,8,type1size,20,320,300,20);
s11.setWindow(GUIwindow);
s11.setDecimalPrecision(1);
s11.addListener(myListener);

Slider s12 = control.addSlider("type2size",0,8,type2size,20,350,300,20);
s12.setWindow(GUIwindow);
s12.setDecimalPrecision(1);
s12.addListener(myListener);

Slider s13 = control.addSlider("type3size",0,8,type3size,20,380,300,20);
s13.setWindow(GUIwindow);
s13.setDecimalPrecision(1);
s13.addListener(myListener);

Slider s14 = control.addSlider("type4size",0,8,type4size,20,410,300,20);
s14.setWindow(GUIwindow);
s14.setDecimalPrecision(1);
s14.addListener(myListener);

Radio r1 = control.addRadio("displayGrid",20,460);

```

```

r1.add("GRID OFF",0);

r1.add("GRID ON",1);

r1.setWindow(GUIwindow);

Radio r2 = control.addRadio("displayCore",100,460);

r2.add("CORE ON",1);

r2.add("CORE OFF",0);

r2.setWindow(GUIwindow);

Radio r3 = control.addRadio("displayColors",180,460);

r3.add("COLORS ON",1);

r3.add("COLORS OFF",0);

r3.setWindow(GUIwindow);

Radio r4 = control.addRadio("isWireframe",260,460);

r4.add("WIREFRAME OFF",0);

r4.add("WIREFRAME ON",1);

r4.setWindow(GUIwindow);

}

class MyControlListener implements ControlListener {

public void controlEvent(ControlEvent theEvent) {

core1 = new Core(new Vec3D(0, 0, 0));

}

}

```

Betik A.10: Kütüphaneler.

```

// LIBRARIES -----

import processing.opengl.*;

import peasy.*;

import toxi.geom.*;

import controlP5.*;

```

Betik A.11: Notepad Çekirdek.

```

_Box -12.0,-12.0,-75.0 r30.0,30.0 150.0

```

Betik A.12: Notepad Tip-1.

_Box -18.0,-12.0,-73.5 r6.0,6.0 3.0
_Box -18.0,-6.0,-73.5 r6.0,6.0 3.0
_Box -12.0,18.0,-73.5 r6.0,6.0 3.0
_Box -6.0,18.0,-73.5 r6.0,6.0 3.0
_Box -12.0,-18.0,-70.5 r6.0,6.0 3.0
_Box -6.0,-18.0,-70.5 r6.0,6.0 3.0
_Box 18.0,12.0,-67.5 r6.0,6.0 3.0
_Box 18.0,18.0,-67.5 r6.0,6.0 3.0
_Box 18.0,6.0,-58.5 r6.0,6.0 3.0
_Box 18.0,12.0,-58.5 r6.0,6.0 3.0
_Box 18.0,-12.0,-55.5 r6.0,6.0 3.0
_Box 18.0,-6.0,-55.5 r6.0,6.0 3.0
_Box 6.0,-18.0,-49.5 r6.0,6.0 3.0
_Box 12.0,-18.0,-49.5 r6.0,6.0 3.0
_Box -24.0,12.0,-40.5 r6.0,6.0 3.0
_Box -18.0,12.0,-40.5 r6.0,6.0 3.0
_Box -12.0,-18.0,-34.5 r6.0,6.0 3.0
_Box -6.0,-18.0,-34.5 r6.0,6.0 3.0
_Box -18.0,-18.0,-31.5 r6.0,6.0 3.0
_Box -12.0,-18.0,-31.5 r6.0,6.0 3.0
_Box -12.0,18.0,-31.5 r6.0,6.0 3.0
_Box -6.0,18.0,-31.5 r6.0,6.0 3.0
_Box 6.0,18.0,-31.5 r6.0,6.0 3.0
_Box 12.0,18.0,-31.5 r6.0,6.0 3.0
_Box -18.0,-18.0,-28.5 r6.0,6.0 3.0
_Box -12.0,-18.0,-28.5 r6.0,6.0 3.0
_Box -12.0,18.0,-28.5 r6.0,6.0 3.0
_Box -12.0,24.0,-28.5 r6.0,6.0 3.0
_Box -18.0,-18.0,-25.5 r6.0,6.0 3.0
_Box -18.0,-12.0,-25.5 r6.0,6.0 3.0
_Box 18.0,-18.0,-25.5 r6.0,6.0 3.0
_Box 18.0,-12.0,-25.5 r6.0,6.0 3.0
_Box 18.0,6.0,-25.5 r6.0,6.0 3.0
_Box 18.0,12.0,-25.5 r6.0,6.0 3.0
_Box -6.0,18.0,-19.5 r6.0,6.0 3.0
_Box -6.0,24.0,-19.5 r6.0,6.0 3.0
_Box 12.0,18.0,-16.5 r6.0,6.0 3.0
_Box 18.0,18.0,-16.5 r6.0,6.0 3.0
_Box -6.0,-24.0,-13.5 r6.0,6.0 3.0
_Box -6.0,-18.0,-13.5 r6.0,6.0 3.0
_Box 18.0,-12.0,-13.5 r6.0,6.0 3.0
_Box 24.0,-12.0,-13.5 r6.0,6.0 3.0
_Box -6.0,18.0,-10.5 r6.0,6.0 3.0
_Box -6.0,24.0,-10.5 r6.0,6.0 3.0
_Box 12.0,-18.0,-4.5 r6.0,6.0 3.0
_Box 18.0,-18.0,-4.5 r6.0,6.0 3.0
_Box -24.0,-6.0,-1.5 r6.0,6.0 3.0
_Box -18.0,-6.0,-1.5 r6.0,6.0 3.0
_Box 6.0,-24.0,4.5 r6.0,6.0 3.0
_Box 6.0,-18.0,4.5 r6.0,6.0 3.0
_Box -18.0,6.0,4.5 r6.0,6.0 3.0
_Box -18.0,12.0,4.5 r6.0,6.0 3.0
_Box -12.0,-18.0,7.5 r6.0,6.0 3.0
_Box -6.0,-18.0,7.5 r6.0,6.0 3.0
_Box 6.0,18.0,10.5 r6.0,6.0 3.0
_Box 6.0,24.0,10.5 r6.0,6.0 3.0
_Box 18.0,6.0,13.5 r6.0,6.0 3.0
_Box 18.0,12.0,13.5 r6.0,6.0 3.0

_Box -24.0,-12.0,16.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,16.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,19.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,19.5 r6.0,6.0 3.0
 _Box 18.0,6.0,22.5 r6.0,6.0 3.0
 _Box 18.0,12.0,22.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,25.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,25.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,28.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,28.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,31.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,31.5 r6.0,6.0 3.0
 _Box 12.0,18.0,37.5 r6.0,6.0 3.0
 _Box 18.0,18.0,37.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,40.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,40.5 r6.0,6.0 3.0
 _Box 6.0,18.0,40.5 r6.0,6.0 3.0
 _Box 12.0,18.0,40.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,46.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,46.5 r6.0,6.0 3.0
 _Box -12.0,18.0,49.5 r6.0,6.0 3.0
 _Box -6.0,18.0,49.5 r6.0,6.0 3.0
 _Box 6.0,18.0,49.5 r6.0,6.0 3.0
 _Box 12.0,18.0,49.5 r6.0,6.0 3.0
 _Box 18.0,6.0,52.5 r6.0,6.0 3.0
 _Box 24.0,6.0,52.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,55.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,55.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,64.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,64.5 r6.0,6.0 3.0
 _Box -24.0,6.0,64.5 r6.0,6.0 3.0
 _Box -18.0,6.0,64.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,67.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,67.5 r6.0,6.0 3.0
 _Box -24.0,6.0,67.5 r6.0,6.0 3.0
 _Box -18.0,6.0,67.5 r6.0,6.0 3.0

Betik A.13: Notepad Tip-2.

_Box -24.0,12.0,-70.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-70.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-70.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-64.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-64.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-64.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-64.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-64.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-64.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-61.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-61.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-61.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-55.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,-55.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-55.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-55.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-55.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-55.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-52.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-52.5 r6.0,6.0 3.0

_Box 18.0,-18.0,-52.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-52.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-52.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-52.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-46.5 r6.0,6.0 3.0
 _Box -24.0,18.0,-46.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-46.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-43.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-43.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-43.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-40.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-40.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,-40.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-37.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-37.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-37.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-22.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-22.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-22.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-22.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-22.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-22.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-19.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-19.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-19.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-16.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-16.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-16.5 r6.0,6.0 3.0
 _Box 18.0,6.0,-13.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-13.5 r6.0,6.0 3.0
 _Box 24.0,12.0,-13.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-10.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-10.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-10.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-7.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-7.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-7.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-4.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-4.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-4.5 r6.0,6.0 3.0
 _Box 18.0,6.0,-4.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-4.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-4.5 r6.0,6.0 3.0
 _Box 18.0,6.0,-1.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-1.5 r6.0,6.0 3.0
 _Box 24.0,12.0,-1.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,1.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,1.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,1.5 r6.0,6.0 3.0
 _Box -24.0,12.0,1.5 r6.0,6.0 3.0
 _Box -24.0,18.0,1.5 r6.0,6.0 3.0
 _Box -18.0,12.0,1.5 r6.0,6.0 3.0
 _Box 6.0,18.0,1.5 r6.0,6.0 3.0
 _Box 6.0,24.0,1.5 r6.0,6.0 3.0
 _Box 12.0,24.0,1.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,4.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,4.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,4.5 r6.0,6.0 3.0
 _Box 6.0,18.0,4.5 r6.0,6.0 3.0
 _Box 12.0,18.0,4.5 r6.0,6.0 3.0

_Box 12.0,24.0,4.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,10.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,10.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,10.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,10.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,10.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,10.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,16.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,16.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,16.5 r6.0,6.0 3.0
 _Box -12.0,18.0,16.5 r6.0,6.0 3.0
 _Box -12.0,24.0,16.5 r6.0,6.0 3.0
 _Box -6.0,24.0,16.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,22.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,22.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,22.5 r6.0,6.0 3.0
 _Box -12.0,24.0,22.5 r6.0,6.0 3.0
 _Box -6.0,18.0,22.5 r6.0,6.0 3.0
 _Box -6.0,24.0,22.5 r6.0,6.0 3.0
 _Box -12.0,18.0,25.5 r6.0,6.0 3.0
 _Box -12.0,24.0,25.5 r6.0,6.0 3.0
 _Box -6.0,18.0,25.5 r6.0,6.0 3.0
 _Box 18.0,6.0,25.5 r6.0,6.0 3.0
 _Box 18.0,12.0,25.5 r6.0,6.0 3.0
 _Box 18.0,18.0,25.5 r6.0,6.0 3.0
 _Box 12.0,18.0,28.5 r6.0,6.0 3.0
 _Box 12.0,24.0,28.5 r6.0,6.0 3.0
 _Box 18.0,24.0,28.5 r6.0,6.0 3.0
 _Box -18.0,24.0,31.5 r6.0,6.0 3.0
 _Box -12.0,18.0,31.5 r6.0,6.0 3.0
 _Box -12.0,24.0,31.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,34.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,34.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,34.5 r6.0,6.0 3.0
 _Box -24.0,12.0,34.5 r6.0,6.0 3.0
 _Box -18.0,6.0,34.5 r6.0,6.0 3.0
 _Box -18.0,12.0,34.5 r6.0,6.0 3.0
 _Box -24.0,18.0,37.5 r6.0,6.0 3.0
 _Box -18.0,18.0,37.5 r6.0,6.0 3.0
 _Box -12.0,18.0,37.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,40.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,40.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,40.5 r6.0,6.0 3.0
 _Box -18.0,6.0,40.5 r6.0,6.0 3.0
 _Box -18.0,12.0,40.5 r6.0,6.0 3.0
 _Box -18.0,18.0,40.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,43.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,43.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,43.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,43.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,43.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,43.5 r6.0,6.0 3.0
 _Box -18.0,18.0,43.5 r6.0,6.0 3.0
 _Box -12.0,18.0,43.5 r6.0,6.0 3.0
 _Box -12.0,24.0,43.5 r6.0,6.0 3.0
 _Box 6.0,18.0,43.5 r6.0,6.0 3.0
 _Box 12.0,18.0,43.5 r6.0,6.0 3.0
 _Box 18.0,18.0,43.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,46.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,46.5 r6.0,6.0 3.0

_Box 12.0,-18.0,46.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,49.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,49.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,49.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,58.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,58.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,58.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,58.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,58.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,58.5 r6.0,6.0 3.0
 _Box -12.0,18.0,58.5 r6.0,6.0 3.0
 _Box -6.0,18.0,58.5 r6.0,6.0 3.0
 _Box -6.0,24.0,58.5 r6.0,6.0 3.0
 _Box 6.0,18.0,61.5 r6.0,6.0 3.0
 _Box 12.0,18.0,61.5 r6.0,6.0 3.0
 _Box 12.0,24.0,61.5 r6.0,6.0 3.0
 _Box 18.0,6.0,67.5 r6.0,6.0 3.0
 _Box 18.0,12.0,67.5 r6.0,6.0 3.0
 _Box 18.0,18.0,67.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,73.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,73.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,73.5 r6.0,6.0 3.0
 _Box -18.0,18.0,73.5 r6.0,6.0 3.0
 _Box -18.0,24.0,73.5 r6.0,6.0 3.0
 _Box -12.0,18.0,73.5 r6.0,6.0 3.0

Betik A.14: Notepad Tip-3.

_Box 18.0,-24.0,-73.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-73.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-73.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-73.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-73.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-73.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-73.5 r6.0,6.0 3.0
 _Box 24.0,12.0,-73.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-67.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,-67.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,-67.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-67.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,-64.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-64.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-64.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-64.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-58.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-58.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,-58.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-58.5 r6.0,6.0 3.0
 _Box -24.0,18.0,-58.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-58.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-58.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-58.5 r6.0,6.0 3.0
 _Box 18.0,6.0,-55.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-55.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-55.5 r6.0,6.0 3.0
 _Box 24.0,6.0,-55.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,-52.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,-52.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,-52.5 r6.0,6.0 3.0

_Box -6.0,-18.0,-52.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-52.5 r6.0,6.0 3.0
 _Box 24.0,6.0,-52.5 r6.0,6.0 3.0
 _Box 24.0,12.0,-52.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-52.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,-46.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-46.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-46.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,-46.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-46.5 r6.0,6.0 3.0
 _Box 6.0,24.0,-46.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-46.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-46.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-43.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-43.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-43.5 r6.0,6.0 3.0
 _Box 18.0,24.0,-43.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,-40.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,-40.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,-40.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-40.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-40.5 r6.0,6.0 3.0
 _Box 6.0,24.0,-40.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-40.5 r6.0,6.0 3.0
 _Box 18.0,24.0,-40.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,-31.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-31.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-31.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-31.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-28.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-28.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-28.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-28.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-22.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-22.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-22.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-22.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-19.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-19.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-19.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-19.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-4.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-4.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-4.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-4.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-1.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-1.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-1.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-1.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,7.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,7.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,7.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,7.5 r6.0,6.0 3.0
 _Box -24.0,18.0,7.5 r6.0,6.0 3.0
 _Box -18.0,12.0,7.5 r6.0,6.0 3.0
 _Box -18.0,18.0,7.5 r6.0,6.0 3.0
 _Box -12.0,18.0,7.5 r6.0,6.0 3.0
 _Box -24.0,18.0,10.5 r6.0,6.0 3.0
 _Box -18.0,12.0,10.5 r6.0,6.0 3.0
 _Box -18.0,18.0,10.5 r6.0,6.0 3.0

_Box -12.0,18.0,10.5 r6.0,6.0 3.0
 _Box -24.0,6.0,13.5 r6.0,6.0 3.0
 _Box -18.0,6.0,13.5 r6.0,6.0 3.0
 _Box -18.0,12.0,13.5 r6.0,6.0 3.0
 _Box -18.0,18.0,13.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,19.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,19.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,19.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,19.5 r6.0,6.0 3.0
 _Box -24.0,6.0,19.5 r6.0,6.0 3.0
 _Box -24.0,12.0,19.5 r6.0,6.0 3.0
 _Box -24.0,18.0,19.5 r6.0,6.0 3.0
 _Box -18.0,6.0,19.5 r6.0,6.0 3.0
 _Box 6.0,18.0,19.5 r6.0,6.0 3.0
 _Box 12.0,18.0,19.5 r6.0,6.0 3.0
 _Box 12.0,24.0,19.5 r6.0,6.0 3.0
 _Box 18.0,24.0,19.5 r6.0,6.0 3.0
 _Box -18.0,18.0,28.5 r6.0,6.0 3.0
 _Box -12.0,18.0,28.5 r6.0,6.0 3.0
 _Box -12.0,24.0,28.5 r6.0,6.0 3.0
 _Box -6.0,18.0,28.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,31.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,31.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,31.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,31.5 r6.0,6.0 3.0
 _Box -18.0,-24.0,34.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,34.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,34.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,34.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,37.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,37.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,37.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,37.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,37.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,37.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,37.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,37.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,52.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,52.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,52.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,52.5 r6.0,6.0 3.0
 _Box -24.0,12.0,52.5 r6.0,6.0 3.0
 _Box -24.0,18.0,52.5 r6.0,6.0 3.0
 _Box -18.0,12.0,52.5 r6.0,6.0 3.0
 _Box -18.0,18.0,52.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,55.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,55.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,55.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,55.5 r6.0,6.0 3.0
 _Box -18.0,12.0,55.5 r6.0,6.0 3.0
 _Box -18.0,18.0,55.5 r6.0,6.0 3.0
 _Box -18.0,24.0,55.5 r6.0,6.0 3.0
 _Box -12.0,18.0,55.5 r6.0,6.0 3.0
 _Box 18.0,6.0,58.5 r6.0,6.0 3.0
 _Box 18.0,12.0,58.5 r6.0,6.0 3.0
 _Box 18.0,18.0,58.5 r6.0,6.0 3.0
 _Box 24.0,12.0,58.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,61.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,61.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,61.5 r6.0,6.0 3.0

_Box -6.0,-18.0,61.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,61.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,61.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,61.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,61.5 r6.0,6.0 3.0
 _Box 6.0,18.0,64.5 r6.0,6.0 3.0
 _Box 6.0,24.0,64.5 r6.0,6.0 3.0
 _Box 12.0,18.0,64.5 r6.0,6.0 3.0
 _Box 12.0,24.0,64.5 r6.0,6.0 3.0
 _Box -18.0,12.0,70.5 r6.0,6.0 3.0
 _Box -18.0,18.0,70.5 r6.0,6.0 3.0
 _Box -18.0,24.0,70.5 r6.0,6.0 3.0
 _Box -12.0,18.0,70.5 r6.0,6.0 3.0
 _Box 6.0,18.0,70.5 r6.0,6.0 3.0
 _Box 12.0,18.0,70.5 r6.0,6.0 3.0
 _Box 12.0,24.0,70.5 r6.0,6.0 3.0
 _Box 18.0,18.0,70.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,73.5 r6.0,6.0 3.0
 _Box 24.0,-24.0,73.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,73.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,73.5 r6.0,6.0 3.0
 _Box 18.0,12.0,73.5 r6.0,6.0 3.0
 _Box 18.0,18.0,73.5 r6.0,6.0 3.0
 _Box 24.0,6.0,73.5 r6.0,6.0 3.0
 _Box 24.0,12.0,73.5 r6.0,6.0 3.0

Betik A.15: Notepad Tip-4.

_Box 18.0,-18.0,-70.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-70.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-70.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-70.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-70.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-70.5 r6.0,6.0 3.0
 _Box 6.0,24.0,-70.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-70.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-70.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-70.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-67.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-67.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,-67.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-67.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-67.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-67.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-67.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-67.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-67.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-67.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-64.5 r6.0,6.0 3.0
 _Box 6.0,24.0,-64.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-64.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-64.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-64.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,-61.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-61.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-61.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-61.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-61.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-61.5 r6.0,6.0 3.0

_Box 18.0,-12.0,-61.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-61.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-61.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-61.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-61.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-61.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-61.5 r6.0,6.0 3.0
 _Box 24.0,12.0,-61.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-61.5 r6.0,6.0 3.0
 _Box -24.0,-24.0,-58.5 r6.0,6.0 3.0
 _Box -18.0,-24.0,-58.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,-58.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,-58.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-58.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,-49.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-49.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-49.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-49.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-49.5 r6.0,6.0 3.0
 _Box -24.0,24.0,-49.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-49.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-49.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-49.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-49.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-49.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-49.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-49.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-49.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-49.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-46.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-46.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-46.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-46.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-46.5 r6.0,6.0 3.0
 _Box -18.0,-24.0,-43.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-43.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,-43.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,-43.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-43.5 r6.0,6.0 3.0
 _Box -24.0,24.0,-43.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-43.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-43.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-43.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-43.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,-37.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-37.5 r6.0,6.0 3.0
 _Box -18.0,-24.0,-37.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-37.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-37.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-37.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-37.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-37.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-37.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-37.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-37.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-37.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-37.5 r6.0,6.0 3.0
 _Box -6.0,18.0,-37.5 r6.0,6.0 3.0
 _Box -6.0,24.0,-37.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-34.5 r6.0,6.0 3.0

_Box 18.0,-12.0,-34.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-34.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-34.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-34.5 r6.0,6.0 3.0
 _Box -24.0,6.0,-34.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-34.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-34.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-34.5 r6.0,6.0 3.0
 _Box -18.0,24.0,-34.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-34.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-34.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-34.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-34.5 r6.0,6.0 3.0
 _Box 24.0,18.0,-34.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,-28.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-28.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-28.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-28.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-28.5 r6.0,6.0 3.0
 _Box -24.0,6.0,-25.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-25.5 r6.0,6.0 3.0
 _Box -24.0,18.0,-25.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-25.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-25.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-22.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-22.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-22.5 r6.0,6.0 3.0
 _Box 18.0,24.0,-22.5 r6.0,6.0 3.0
 _Box 24.0,24.0,-22.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-19.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-19.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,-19.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-19.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-19.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,-16.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,-16.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,-16.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,-16.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,-16.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,-16.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,-16.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,-16.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-16.5 r6.0,6.0 3.0
 _Box 24.0,-6.0,-16.5 r6.0,6.0 3.0
 _Box -24.0,6.0,-13.5 r6.0,6.0 3.0
 _Box -24.0,12.0,-13.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-13.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-13.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-13.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-10.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-10.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-10.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,-10.5 r6.0,6.0 3.0
 _Box 24.0,-12.0,-10.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-10.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-10.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-10.5 r6.0,6.0 3.0
 _Box 18.0,12.0,-10.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-10.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,-7.5 r6.0,6.0 3.0

_Box -18.0,-18.0,-7.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,-7.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,-7.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,-7.5 r6.0,6.0 3.0
 _Box -18.0,6.0,-7.5 r6.0,6.0 3.0
 _Box -18.0,12.0,-7.5 r6.0,6.0 3.0
 _Box -18.0,18.0,-7.5 r6.0,6.0 3.0
 _Box -12.0,18.0,-7.5 r6.0,6.0 3.0
 _Box -12.0,24.0,-7.5 r6.0,6.0 3.0
 _Box 6.0,18.0,-7.5 r6.0,6.0 3.0
 _Box 6.0,24.0,-7.5 r6.0,6.0 3.0
 _Box 12.0,18.0,-7.5 r6.0,6.0 3.0
 _Box 12.0,24.0,-7.5 r6.0,6.0 3.0
 _Box 18.0,18.0,-7.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,-1.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,-1.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,-1.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,-1.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,-1.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,1.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,1.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,1.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,1.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,1.5 r6.0,6.0 3.0
 _Box 18.0,6.0,7.5 r6.0,6.0 3.0
 _Box 18.0,12.0,7.5 r6.0,6.0 3.0
 _Box 24.0,6.0,7.5 r6.0,6.0 3.0
 _Box 24.0,12.0,7.5 r6.0,6.0 3.0
 _Box 24.0,18.0,7.5 r6.0,6.0 3.0
 _Box -24.0,-24.0,13.5 r6.0,6.0 3.0
 _Box -18.0,-24.0,13.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,13.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,13.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,13.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,13.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,13.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,13.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,13.5 r6.0,6.0 3.0
 _Box 24.0,-24.0,13.5 r6.0,6.0 3.0
 _Box 12.0,18.0,16.5 r6.0,6.0 3.0
 _Box 18.0,6.0,16.5 r6.0,6.0 3.0
 _Box 18.0,12.0,16.5 r6.0,6.0 3.0
 _Box 18.0,18.0,16.5 r6.0,6.0 3.0
 _Box 24.0,12.0,16.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,22.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,22.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,22.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,22.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,22.5 r6.0,6.0 3.0
 _Box -24.0,-18.0,25.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,25.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,25.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,25.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,25.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,28.5 r6.0,6.0 3.0
 _Box -12.0,-24.0,28.5 r6.0,6.0 3.0
 _Box -12.0,-18.0,28.5 r6.0,6.0 3.0
 _Box -6.0,-24.0,28.5 r6.0,6.0 3.0
 _Box -6.0,-18.0,28.5 r6.0,6.0 3.0
 _Box 6.0,18.0,31.5 r6.0,6.0 3.0

_Box 6.0,24.0,31.5 r6.0,6.0 3.0
 _Box 12.0,18.0,31.5 r6.0,6.0 3.0
 _Box 18.0,12.0,31.5 r6.0,6.0 3.0
 _Box 18.0,18.0,31.5 r6.0,6.0 3.0
 _Box 6.0,18.0,34.5 r6.0,6.0 3.0
 _Box 6.0,24.0,34.5 r6.0,6.0 3.0
 _Box 12.0,18.0,34.5 r6.0,6.0 3.0
 _Box 12.0,24.0,34.5 r6.0,6.0 3.0
 _Box 18.0,18.0,34.5 r6.0,6.0 3.0
 _Box -24.0,6.0,46.5 r6.0,6.0 3.0
 _Box -24.0,12.0,46.5 r6.0,6.0 3.0
 _Box -18.0,6.0,46.5 r6.0,6.0 3.0
 _Box -18.0,12.0,46.5 r6.0,6.0 3.0
 _Box -18.0,18.0,46.5 r6.0,6.0 3.0
 _Box 6.0,18.0,46.5 r6.0,6.0 3.0
 _Box 12.0,18.0,46.5 r6.0,6.0 3.0
 _Box 12.0,24.0,46.5 r6.0,6.0 3.0
 _Box 18.0,18.0,46.5 r6.0,6.0 3.0
 _Box 24.0,18.0,46.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,49.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,49.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,49.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,49.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,49.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,52.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,52.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,52.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,52.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,52.5 r6.0,6.0 3.0
 _Box 18.0,6.0,55.5 r6.0,6.0 3.0
 _Box 18.0,12.0,55.5 r6.0,6.0 3.0
 _Box 18.0,18.0,55.5 r6.0,6.0 3.0
 _Box 18.0,24.0,55.5 r6.0,6.0 3.0
 _Box 24.0,6.0,55.5 r6.0,6.0 3.0
 _Box -24.0,12.0,61.5 r6.0,6.0 3.0
 _Box -18.0,6.0,61.5 r6.0,6.0 3.0
 _Box -18.0,12.0,61.5 r6.0,6.0 3.0
 _Box -18.0,18.0,61.5 r6.0,6.0 3.0
 _Box -18.0,24.0,61.5 r6.0,6.0 3.0
 _Box 6.0,-24.0,64.5 r6.0,6.0 3.0
 _Box 6.0,-18.0,64.5 r6.0,6.0 3.0
 _Box 12.0,-18.0,64.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,64.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,64.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,67.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,67.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,67.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,67.5 r6.0,6.0 3.0
 _Box 24.0,-18.0,67.5 r6.0,6.0 3.0
 _Box -24.0,-12.0,70.5 r6.0,6.0 3.0
 _Box -24.0,-6.0,70.5 r6.0,6.0 3.0
 _Box -18.0,-18.0,70.5 r6.0,6.0 3.0
 _Box -18.0,-12.0,70.5 r6.0,6.0 3.0
 _Box -18.0,-6.0,70.5 r6.0,6.0 3.0
 _Box 12.0,-24.0,70.5 r6.0,6.0 3.0
 _Box 18.0,-24.0,70.5 r6.0,6.0 3.0
 _Box 18.0,-18.0,70.5 r6.0,6.0 3.0
 _Box 18.0,-12.0,70.5 r6.0,6.0 3.0
 _Box 18.0,-6.0,70.5 r6.0,6.0 3.0

ÖZGEÇMİŞ



Ad Soyad: Burak Yardımcı

Doğum Yeri ve Tarihi: Bursa 30.04.1987

Adres: Acıbadem Mah. Günel Sk. Gülevler Sit. 5. Blok Daire 28 Üsküdar İstanbul

E-Posta: burakyardimci29@gmail.com

Lisans: İstanbul Teknik Üniversitesi Mimarlık Fakültesi

Mesleki Deneyim ve Ödüller:

2012 – Halen: Adeas Mimarlık

2011 – 2012: DS Mimarlık

2009 – 2011: TMO Mimarlık